Learning Enabled Optimization: Towards a Fusion of Statistical Learning and Stochastic Optimization

Suvrajeet Sen, Yunxiao Deng
Daniel J. Epstein Department of Industrial and Systems Engineering
University of Southern California, Los Angeles, 90089
s.sen@usc.edu, yunxiaod@usc.edu

Several emerging applications, such as "Analytics of Things" and "Integrative Analytics" call for a fusion of statistical learning (SL) and stochastic optimization (SO). The Learning Enabled Optimization paradigm fuses concepts from these disciplines in a manner which not only enriches both SL and SO, but also provides a framework which supports rapid model updates and optimization, together with a methodology for rapid model-validation, assessment, and selection. Moreover, in many big data/big decisions applications these steps are repetitive, and possible only through a continuous cycle involving data analysis, optimization, and validation. This paper sets forth the foundation for such a framework by introducing several novel concepts such as *statistical optimality, hypothesis tests for model-fidelity, generalization error of stochastic optimization,* and finally, a *non-parametric methodology for model selection.* These new concepts provide a formal framework for modeling, solving, validating, and reporting solutions for Learning Enabled Optimization (LEO). We illustrate the LEO framework by applying it to an inventory control model in which we use demand data available for ARIMA modeling in the statistical package "R". In addition, we also study a production-marketing coordination model based on combining a pedagogical production planning model with an advertising data set intended for sales prediction. Because the approach requires the solution of several stochastic programming instances, some using continuous random variables, we leverage stochastic decomposition (SD) for the fusion of regression and stochastic linear programming. In this sense, the novelty of this paper is its framework, rather than a specific new algorithm. Finally, we present an architecture of a software framework to bring about the fusion we envision.

Key words: Stochastic Linear Programming, Stochastic Decomposition, Statistical Learning, Model Assessment

---

2017/04/02

1. Introduction

In recent years, optimization algorithms have become the work-horse of statistical (or machine) learning. Whether studying classification using linear/quadratic programming for support vector machines (SVM) or logistic regression using a specialized version of Newton's methods (e.g., for expectation maximization), deterministic optimization algorithms have provided a strong foundation for statistical learning. Indeed, statistical learning could be labeled as "optimization enabled learning". The class of models studied in this paper, entitled Learning Enabled Optimization (LEO), is intended to support stochastic optimization methods which leverage advances in statistical learning. We expect this new approach to be particularly powerful for environments with many data sources (volume), rapid information flow (velocity), and requiring adaptation to uncertain shifts in data (volatility). The proposed fusion is based on an array of new models, methods, and applications with the potential to transform the landscape of OR models and methods. We emphasize that our setting is very different from Statistical Decision Theory whose purpose is to support choices in the context of statistical models. In contrast, the LEO paradigm is intended to support choices in the context of decisions for operations research models in management and engineering.

In terms of scientific genealogy, one can trace the introduction of learning into optimization from the work on approximate dynamic programming (ADP, Bertsekas (2012), Powell (2011)) and approximate linear programming (ALP, e.g. De Farias and Van Roy (2004)). The canonical structure of these approaches pertains to DP, where one uses approximations of the DP value function by using basis functions. In this paper, the canonical setup derives from constrained optimization, although we will state our objectives in the context of approximate solutions. In this sense, one may refer to the technical content of our approach as "approximate constrained stochastic optimization".

This paper is organized as follows. This introductory section consists of two further subsections: one on "Applications of the Future", and another on "Connections to the Literature and Contributions". In section 2, we present two fundamental structures, which we refer to as "LEO Models with Disjoint Spaces" and "LEO Models with Shared Spaces". We illustrate the first of these structures with an inventory control problem, and the second one is illustrated using a production-marketing coordination model. Because LEO models will allow both continuous and discrete random variables (rvs), the statement of

optimization will be relaxed to seek $\delta-$optimum solutions with a high level of reliability (greater than 95%, say). This type of solution is sometimes referred to as a "distribution-free" estimate of the probability of $\delta-$optimality. This concept, which is set forth in section 3, will be referred to as "statistical optimality" for online algorithms (such as Stochastic Decomposition (SD)). In section 4 we study hypothesis tests for model validation, as well as a non-parametric ANOVA which identifies the contenders (models) which may be most promising. In addition, we also define a concept of generalization error which is motivated by an analogous concept in statistical learning. For LEO models, this measure aims to quantify the degree of flexibility expected in the decision model. This entire protocol is illustrated in section 5 via computations for the examples introduced in 2. Finally, section 6 presents our conclusions and a possible path forward for this new genre of models.

1.1. Applications of the Future

In this section, we briefly discuss a couple of applications which are currently on the horizon, and how the new paradigm might help transform the vision of these applications into reality. Needless to say that we are not at the point where we can illustrate the power of our concepts on these applications. However, we will provide examples which will illustrate the paradigm. The two forward-looking applications we have in mind involve "Analytics of Things" and "Integrative Analytics".

Analytics of Things (AoT)

The explosion of sensors and communications devices have enabled things such as home appliances, automobiles, jet engines, and many others to communicate with other devices through the Internet of Things (IoT). For instance in the electricity grid, voltage and phase-angle measurements can be communicated to "hot-start" DC power flow (linear) approximations, allowing the linear approximation to track the nonlinear system. One particularly relevant example of AoT arises in the operation of renewable generators (e.g. solar panels, wind turbines) whose intermittent power production is a fundamental barrier to introducing them into the electricity grid, without a loss in system reliability. With the advent of the IoT, it is conceivable that rapid state updates can be leveraged to allow timely decision-making for currently "undispatchable" (renewable) generators. Most states in the U.S. have mandated renewable portfolio standards, with some states being more aggressive than others. The state of California has passed a law requiring that 50% of all generation be attributed to renewable energy by the year 2030. Recent studies

(Olson et al. (2015)) suggest that the largest integration challenge (using current planning tools) is the pervasive over-generation of power when renewable penetration exceeds 33%. Some of this is due to the need for greater fast back-up generation in cases of higher renewable penetration. Adaptive electricity generation decisions are prospective (because of uncertainty), and must accommodate forecast error, and ramping constraints of the fleet of dispatchable (non-renewable) resources. These challenges cannot be addressed simply by faster processing of wind data; it involves making constrained operational decisions at relatively short time-intervals, and under limited human supervision (see Gangammanavar et al. (2016)). For such settings, the well-known approach of scenario-based stochastic programming (SP) remains one that calls for a great deal of human intervention to assess questions like how many scenarios to use, how should they be generated, which of these should one use etc.? For applications like AoT, models have to be instantiated, validated and updated at time intervals that would make a labor-intensive scenario generation process difficult to support. In addition, the SP literature provides little or no guidance on how model-assessment and selection should be included in such "rapid-fire" environments. In the context of AoT, one expects decisions to be generated in response to evolving observations (e.g., wind speed and direction), and model validation will refer to the ability to provide responsive decision support in swiftly changing regimes. Note that such optimization models need to be assessed for their flexibility as in statistical learning. In order to conduct timely analytics using the IoT, the LEO approach suggests using statistical learning to identify systematic trends and "error-bars" within stochastic optimization. The resulting fusion of learning and stochastic optimization, which leverages IoT, is what we are referring to as AoT.

Integrative Analytics (IA)

A common view of analytics, as summarized by Gartner Analytics, is shown in Figure 1. While this is an insightful classification of different aspects of analytics, it also highlights some of the challenges associated with higher ends of the skills-and-value spectrum. Due to the degree of specialization required in Predictive and Prescriptive Analytics, these areas tend to operate within their own silos. As a result, transforming insight into action is not as straightforward as one may be led to believe. In a recent survey (KPMG 2014) over one hundred CFOs and CIOs of large organizations (over a billion dollars in annual

turnover) were interviewed. Over 96% of those surveyed acknowledged that they could do better with big data, and make better use of analytics. Two of the top three significant questions which emerged from the KPMG survey relate to: a) How will predictive analytics drive future decision making? and b) What technology will be required to operationalize data and analytics within the organization? We address these very questions, albeit, in a manner that goes to the core knowledge gap which exists today: i) how should data sets for statistical learning (or predictive models) be integrated to support decisions from optimization models? and ii) what formal OR support can we provide so that the path forward (i.e. decisions) can be recommended with a quantifiable degree of confidence? We offer the correspondence (a) – (i) and (b) – (ii), and while the former aligns reasonably well, the latter calls for a little clarification: we believe that OR models and software will eventually provide the technology which operationalizes decisions (based on data and analytics). This paper is devoted to the science which will support IA.

1.2. Connections to the Literature and Contributions

In keeping with the goals to accomplish more with data in OR/MS applications, there have been some attempts to have optimization methods guide information gathering for predictive analytics in the work of Frazier (2012) and Ryzhov et al. (2012) which are intended to help an experimentalist improve the effectiveness of predictive models by using sensitivity of the response (using a concept known as knowledge gradient) to design experiments. This line of work uses algorithmic notions of optimization for experimental design (including simulation experiments). A slightly different viewpoint at the interface between predictive and prescriptive analytics is the recent working paper by Bertsimas and Kallus (2014) where the authors show how Predictive Analytics (i.e. forecasting models) can be used to enhance outcomes of Prescriptive Analytics by allowing more accurate forecasts to guide prescriptions. Thus, instead of addressing a fusion of these aspects of analytics, their goal is to improve prescriptive analytics by using model-based forecasts. They show that a model-free approach (which the authors refer to as Sample Average Approximation (SAA)) can lead to poor decisions. However, we should observe that SAA by itself is general enough to accommodate model-based forecasts in many instances (see Sen et al. (2006)), including the examples mentioned in this paper.

As for the OR/MS literature, the primary focus of statistical learning has been on specific classes of problems (e.g. newsvendor models). For instance, Liyanage and Shanthikumar

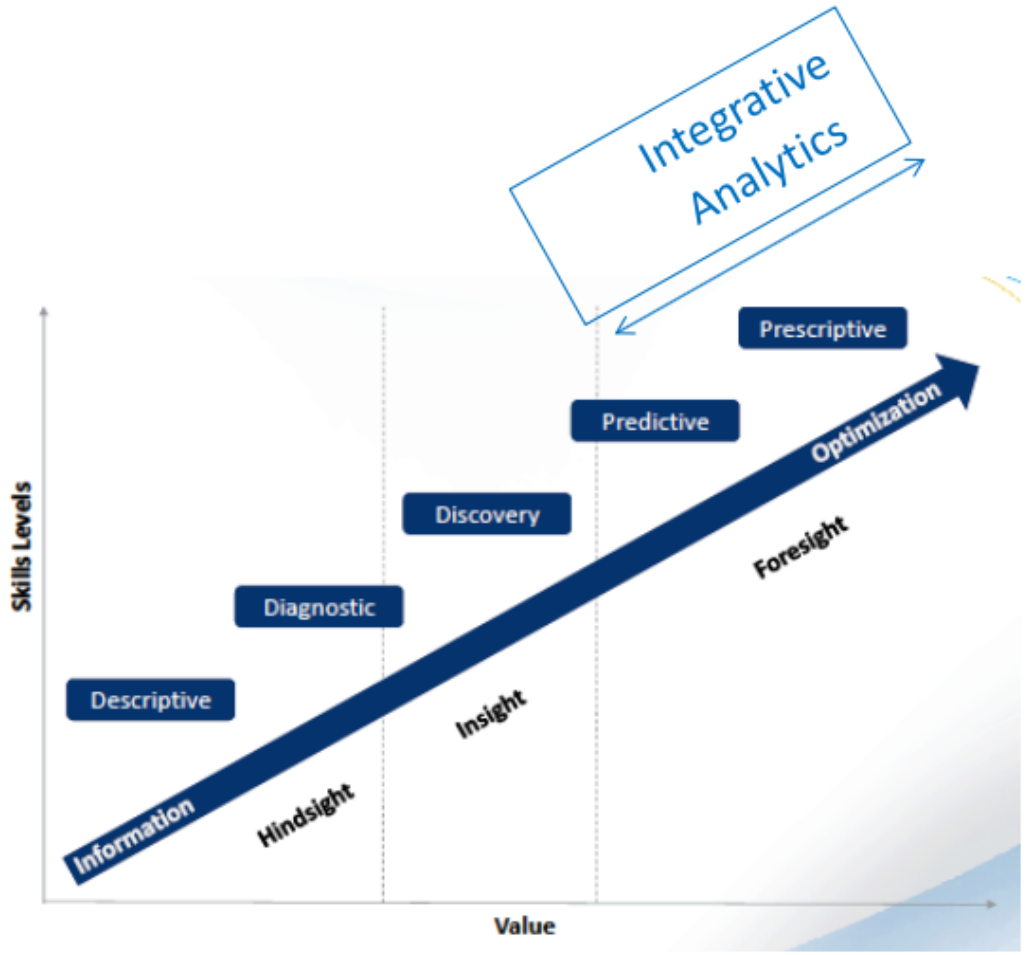

Figure 1. Five Types of Analytics (Source: Gartner Analytics).

Figure 1 **Types/Phases of Analytics. Source: Gartner Analytics. Integrative Analytics box added by the authors.**

(2005) and more recently Rudin and Vahn (2014) have studied the integration of optimization and learning to identify optimal inventory ordering policies. Indeed the former paper has demonstrated, using a simple newsvendor example, that combining statistical estimation/forecasting with optimization provides superior policies, when compared with a model in which estimation and optimization are carried out separately. However, the approach of that paper relies heavily on the simplicity of optimization involved in newsvendor models, and and moreover, distributional assumptions are critical. On the other hand, the latter paper (Rudin and Vahn (2014)) does view the newsvendor decisions in a statistical/machine learning setting. Although their setup views both estimations and decisions

within one model, our approach subsumes theirs because our optimization model is so much more general. Indeed, the (NV-reg) model of Rudin and Vahn (2014) can be solved by using their regression coefficients (q) in the first stage and the overage/shortage variables of newsvendor models as second stage variables. Nevertheless, such use of our setup may violate the assumptions we impose on errors associated with the statistical model.

As the reader will recognize from our paper, our approach allows very general decision models. However, unlike most previous approaches, including stochastic programming (SP), the LEO approach is based on choosing the most promising model from among a collection of alternatives which seem promising in a learning process. For each model-type, we will carry out a collection of tests both before and after optimization to help guide model-choice. In this context, we suggest statistical estimates, and tests which support this choice. In particular, we provide

- a distribution-free procedure to estimate of the probability of $\delta-$optimality, which we refer to as "statistical optimality",
- a collection of hypothesis tests which help assess model-fidelity,
- the notion of generalization error in the context of stochastic optimization,
- a non-parametric ANOVA approach to verify whether the results of one model-type dominate those of another, thus ultimately suggesting the most promising model.

The novelty of these contributions is self-evident, not only from a conceptual (i.e. theoretical) point of view, but also from modeling and computational perspectives. Using examples from the OR/MS discipline, we show how these ideas provide decision support which combines both statistical learning as well as stochastic optimization. While our examples are drawn from the simplest class of models[1], we believe that the main contributions listed above have the potential to change the future of OR/MS modeling, especially for cases in which systematic patterns can be discovered and utilized via a combination of SL and SO tools.

2. Learning Enabled Optimization

As illustrated in Figure 1, Statistical Learning provides support for predictive analytics, whereas, optimization forms the basis for prescriptive analytics, and the methodologies for these are built independently of each other. The process recommended for SL is

[1] combining multiple linear regression and stochastic linear programming

summarized in Figure 2a in which the entire data set is divided into two parts (Training and Validation), with the former being used to learn model parameters (for a predictive model), and the latter data set used for model assessment and selection. Once a model is selected, it can be finally tested via either simulation or using an additional "test data set" for trials before adoption. This last phase is not depicted in Figure 2 because the concepts for that phase can mimic those from the model validation phase.

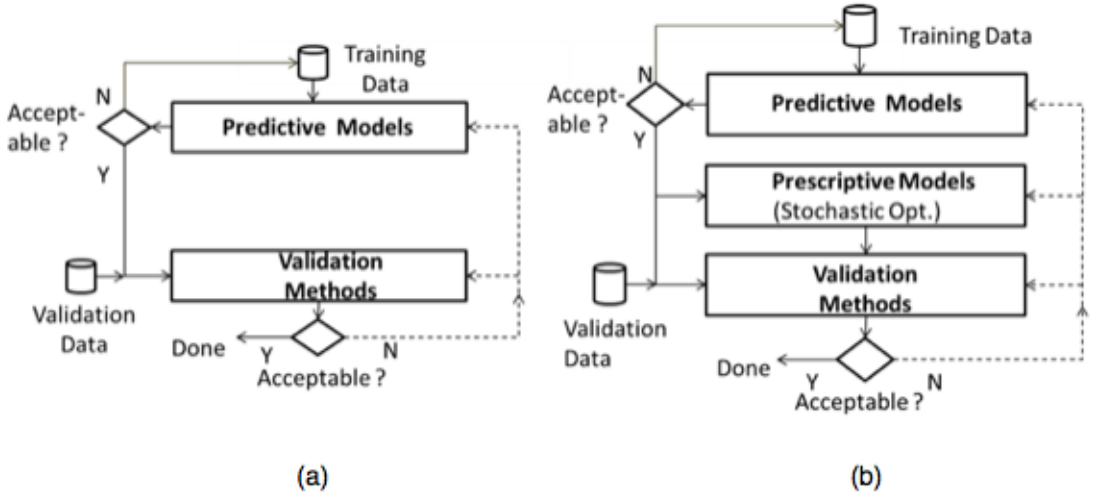

Figure 2 Statistical Learning and Learning Enabled Optimization

2.1. Model Instantiation

The main theme of this section involves stating our aspirations for LEO models. As one might expect, this framework consists of two major parts: the SL piece and the SO piece. We begin by stating a regression model in its relatively standard form. Towards this end, let m denote an arbitrary regression model for a training set of observations $\{W_i, Z_i\}$, indexed by $i \in T$, the training data. For notational simplicity we assume that $W_i \in \mathbb{R}$, whereas $Z_i \in \mathbb{R}^p$. Given the training data, a class of models $\mathcal{M}$, and a loss function ℓ, the regression is represented as follows:

$$\hat{m} \in \operatorname{argmin}\Big\{\frac{1}{|T|}\sum_{i\in T}\ell(m) \mid m \in \mathcal{M}\Big\}. \tag{1}$$

We wish to emphasize that in many circumstances (e.g., modeling the impact of natural phenomena such as earthquakes, hurricanes etc.), model fidelity may be enhanced by

building the statistical model of the phenomena independently of decision models. For such applications, one may prefer to treat SL and SO independently.

Remark 1. Because of the plausibility of allowing alternative statistical models, it is not unusual for SL models to be chosen from a finite list consisting of potentially justifiable models. We do acknowledge that such a strategy of exploring alternative statistical models could be computationally intensive. Fortunately, there has been significant progress in computational technologies for certain classes of SO models (e.g. Stochastic LPs) where software speed-ups have resulted in gains which outpace Moore's law (for hardware) using processors of roughly the same speed (Sen and Liu (2016)). This also provides the impetus to study algorithms for more general (nonlinear) models. In any event, a list of potential alternative models will be indexed by the letter q, and we suggest representing the class of models by $(\ell_q, \mathcal{M}_q)$, and the specific model obtained in (1) is denoted $\hat{m}_q$. Whenever the specific index of a model is not important, we will simply refer to model as $\hat{m}$. ■

Scalar and Vector Errors

In SL it is not uncommon to postulate a deterministic model $\hat{m}$ together with a collection of scalar error outcomes $\xi_i = W_i - \hat{m}(Z_i), i \in T$ (Hastie et al. (2011)). Because empirical distributions depend on the training set T, the corresponding model will be denoted $\hat{m}_T$, and the error rv will be denoted $\tilde{\xi}_T$[2]. Using these outcomes as the support for an error rv (with weights $\frac{1}{|T|}$), one obtains an empirical distribution $\mathcal{P}_T$. Other models of error are clearly possible by choosing alternative model-types $\hat{m}_q \in \mathcal{M}_q$, such that the distribution functions $\mathcal{P}_q$ satisfy $||\mathcal{P}_q - \mathcal{P}_T|| \leq \Delta_{\mathcal{P}},\ \Delta_{\mathcal{P}} > 0$. For instance, when using multiple linear regression (MLR), one might be prompted to use alternative Gaussian rvs for the regression coefficients. Letting $\hat{\mathbb{P}}$ denote a collection of alternative (plausible) distribution functions for the regression coefficients, we let $\mathbb{P} = \hat{\mathbb{P}} \cup \{\mathcal{P}_T\}$. To give the reader a preview of how alternative models will be assessed (see also section 4), we note that models will be associated with two types of errors: a generalization error to estimate the flexibility of a statistical model, and an estimated optimization error. By choosing a model which balances these two types of errors, one can choose the most acceptable model. Our development relies on the following main assumptions. One additional assumption will be imposed in section 2.3.

[2] When a rv is denoted by a greek letter, there will be a tilde above the letter. Otherwise, a rv will have outcomes that are lowercase and the corresponding uppercase letter will denote the rv.

Assumption 1 (A1). a) First consider scalar (additive) errors defined by $\xi_i := W_i - \hat{m}(Z_i)$. *The error rvs are assumed to be independent and the error distribution does not depend on any specific choice of* $Z = z$. *Thus for the case of scalar (additive) errors, the randomized response* $m(z,\xi) = \hat{m}(z) + \xi$ *(homoscedasticity).*

b) Alternatively, one may define $\hat{m}(z) = \sum_{\tau\in\mathcal{T}} \phi_\tau(\beta_\tau^\top z)$, *where* $\mathcal{T}$ *is a finite index set and* $\phi_\tau : \mathbb{R} \to \mathbb{R}$. *As in a),* $m(z,\xi) = \hat{m}(z) + \xi$, *and the errors are assumed to have the same homoscedasticity properties. This approach leads to a rather general setting due to a result of Diaconis and Shahshahani (1984), which suggests that nonlinear functions of linear combinations can produce arbitrarily close approximations of smooth nonlinear functions.*

c) Finally, consider the case in which the regression coefficients are allowed to be random. In this case, we have vector errors in the following way. Define functions $m(z,\xi) = \sum_\tau \tilde{\beta}_\tau \phi_\tau(z)$, *where* $\phi_0 = 1$, *and* $\phi_\tau(\cdot)$, *are deterministic functions, but the parameters* $\tilde{\beta}_\tau$ *are random variables. Let* $\hat{m}(z) = \sum_\tau \mathbb{E}(\beta_\tau)\phi_\tau(z)$. *In this case, a vector of errors are associated with an outcome of random coefficients* $\{\tilde{\beta}_\tau\}$ *is given by the difference* $\tilde{\xi}_\tau = \tilde{\beta}_\tau - \mathbb{E}(\beta_\tau)$. *The parameters* $\{\beta_\tau\}$ *may be correlated, but the distributions of random coerricients* $\{\tilde{\beta}_\tau\}$ *do not depend on any specific choice of* $Z = z$ *(homoscedasticity). For the case of multiple linear regression, one lets the indices* τ *have a one-to-one correspondence with the index of predictor variables* j, *and then* $\phi_j(z) = z_j$. *In any event, using vector errors* ξ_τ *as defined above, we have* $m(z,\xi) = \hat{m}(z) + \sum_\tau \xi_\tau \phi_\tau(z)$.

Assumption 2 (A2). We will assume that decisions in SO, denoted x, *have no impact on the continuing data process* $\{(W,Z)\}$ *to be observed in the future.*

To put this assumption in the context of some applications, note that assumption A2 holds for the wind-energy application mentioned earlier because decisions to use wind energy do not change the wind processes. Similarly, in the advertising/financial market, it may be assumed that an individual advertiser/investor may not be large enough to change future market conditions.

We will present two alternative structures for the SO part of LEO: a model with "Disjoint Spaces," and another which we refer to as a model with "Shared Spaces". These two structures are presented next.

2.2. LEO Model with Disjoint Spaces

This is the simplest version of a LEO model in which the values of the statistical inputs (denoted Z) do not assume values in the space of optimization variables (x) of the SO model. To motivate this structure, consider the following example.

Example 1. Inventory Control: LEO-`ELECEQUIP` Data

The `ELECEQUIP` data-set in R provides 10 years of demand data for electrical equipment. We present an inventory control model with this data-set in Appendix A. Consider making equipment ordering choices in period t based on demand data from previous periods (i.e., periods $j < t$). Since the optimization model chooses decisions for periods $j \geq t$, we treat the optimization variables and the statistical model as disjoint. Clearly, this property holds for rolling horizon models as well. In essence the disjoint spaces allows the statistical and optimization models to operate by simply passing values assumed by rvs. Because of the disjoint spaces, such a LEO model can entertain reasonably complex descriptions of data (e.g. time series, nonlinear and the so-called link functions). Our preliminary results provide an example using an ARIMA model, with $(p, d, q), (P, D, Q) = (0, 0, 0), (1, 1, 0)$. This example will emphasize the power of a LEO model with disjoint spaces. ■

Let $x \in \mathbf{X} \subseteq \mathbb{R}^{n_1}$ denote the optimization variables, and suppose that the parameters Z have p elements indexed by the set $\mathcal{J} = \{1, ..., p\}$. Let $\tilde{\xi}_q$ denote a rv with distribution $\mathcal{P}_q \in \hat{\mathbb{P}}$. Then a decision model denoted (SO(q)) is given by:

$$x_q \in \delta - \arg\min\{f(x) := c(x) + \mathbb{E}_\xi[H(x, \tilde{\xi}_q | Z = z)], \text{s.t: } x \in \mathbf{X}\} \tag{2}$$

where, $\delta > 0$, and the rv H has outcomes h as defined below.

$$h(x, \xi_q | Z = z) = \min\{d(y) | g(x, y) - m_q(z, \xi_q) \leq 0, y \in \mathbf{Y} \subseteq \mathbb{R}^{n_2}\} \tag{3}$$

Here $g : \mathbb{R}^{n_1+n_2} \to \mathbb{R}$. Clearly these constraints could be multi-dimensional, but some of the same conceptual challenges would persist. For instance as mentioned above, there may be continuous rvs, and an associated distribution in the set $\mathbb{P}$. Finally, we should clarify that the expectation in (2) is calculated with respect to the rv $\tilde{\xi}_q$.

The value function h defined in (3) goes by several alternative names such a "recourse" function in SO or "cost-to-go" function in DP. While the underpinnings of LEO models are closer to SO than DP, we adopt the "cost-to-go" terminology because we plan to extend LEO models to allow other types of cost predictions in the future (e.g., statistical forecasts, computational times and other elements of an uncertain future).

Remark 2. Due to the presence of continuous rvs (or discrete rvs with countably infinite outcomes), it may not be realistic to guarantee a deterministic certificate of $\delta-$optimality.

In such instances, we will relax the deterministic requirement in (2). *We will seek a statistically estimated bound* $\gamma := Prob\ (x_q \in \delta - \arg\min\{f(x)|x \in \mathbf{X}\})$*, and the solution will be deemed acceptable if* $\gamma \geq 0.95$[3]. We will address this question in much greater depth in section 3 and report values of γ and δ in section 5. ■

2.3. LEO Models with Shared Spaces

We continue with assumptions A1 and A2, and will include another assumption for this class of models. Consider a SO model whose decisions x, have a subset of variables $(x_r, r \in J \subset \mathcal{J})$ which take values in the same space as the predictor data. We refer to these variables as ones that share the same spaces, and hence these models will be referred to as "model with Shared Spaces". Let us state a LEO model with Shared Spaces in the following form.

$$x_q \in \delta - \arg\min\{f(x) := c(x) + \mathbb{E}_\xi[H(x, \tilde{\xi}_q|Z = z, z_j = x_j, j \in J)], \text{s.t: } x \in \mathbf{X}\} \tag{4}$$

where, $\delta > 0$, and the rv H has outcomes h as defined below.

$$h(x, \xi_q|Z = z, z_j = x_j, j \in J) := \min\{d(y)|g(x, y) - m_q(z, \xi_q) \leq 0, y \in \mathbf{Y} \subseteq \mathbb{R}^{n_2}\} \tag{5}$$

As in Remark 2, we will seek a statistically estimated bound of $\gamma = Prob\ (x_q \in \delta - \arg\min\{f(x)|x \in \mathbf{X}\})$*, and accept the solution if* $\gamma \geq 0.95$.

Note that in this form of LEO, the decision maker is called upon to make a "bet" $(x_j, j \in J)$, and the response is a rv $H(x, \tilde{\xi}_q|Z = z, z_j = x_j, j \in J)$. Thus, the response of both Disjoint and Shared Spaces models have a similar form, although the decisions play different roles. To accommodate this, we state the following assumption.

Assumption 3 (A3). When decisions x *are allowed to assume values in a subspace of observations of the rv* Z*, we assume that* $\mathbf{X}$ *is a subset of* $\Pi_J\,(conv\ \{Z_{\cdot,1}, \ldots, Z_{\cdot,p}\})$*, where the notation* $\Pi_J(\cdot)$ *denotes the projection on to the subspace of variables indexed by* J*. In the event the regression has identified outliers (as in robust regression), appropriate points should be removed from the data set.*

This assumption will be enforced by our procedures. In order to give the reader a concrete example, we present the following.

Example 2. Production-Marketing Coordination: LEO-Wyndor Data

[3] The choice of threshold 0.95 ensures a reasonably high reliability although other levels are clearly allowed.

An important piece of data for production planning is predicted sales, which in turn depends on how much advertising is carried out. Suppose that a company spends its advertising budget on several media channels, then, this decision has an impact on future sales figures. Appendix A presents a "toy" example which we refer to as the LEO-Wyndor data in which the decision vector x represents the allocation of the advertising budget to each type of media (TV and radio). The name Wyndor and the production part of this problem is borrowed from a very popular textbook (Hillier and Lieberman (2012)). There, this problem is a pedagogical example for LP-based production planning. Our example (see Appendix A) extends the original Wyndor model to one in which the production plan is to be made while bearing in mind that the allocation of the advertising budget affects sales of Wyndor products (two types of doors), and the final production plan will be guided by firm orders (i.e.,sales) in the future. For this example, a statistical model predicting future sales is borrowed from the advertising data set of James et al. (2013) which also presents an MLR model relating sales (W) with advertising expenditures (Z) in TV and radio. The model and data set are summarized in Appendix A. In this example, the advertising decisions constitute a "bet" on the first stage (advertising) decisions x, and the second stage decisions are the production planning choices, given "firm orders" (sales). ■

The LEO models presented above are relatively general, allowing very general regression models such as kernel-based methods, projection pursuit, and others. However, our current computational infrastructure is limited to stochastic linear programming (SLP) and as a result the regression used for models with Shared Spaces will be restricted to MLR. For this reason, we also focus on assumptions A1-a and A1-c, but postpone examples allowing A1-b to a future paper. In case of the Disjoint Spaces however, more general models are permitted but once again, the optimization aspect will be limited to SLP for computational illustrations.

3. Statistical Optimality

Generally speaking, optimization models seek deterministic guarantees of optimality. Exceptions to this observation are typically found in heuristic methods which seek performance guarantees in the form of a deterministic error bound (from optimality). There is a long history of the "curse of dimensionality" in SO, and it is no surprise that the phenomenon also plagues the SL literature. To be sure, statistical optimality bounds have

been studied in the SO literature for a while (e.g., Higle and Sen (1991), Higle and Sen (1996b), Mak et al. (1999), Kleywegt et al. (2002), Bayraksan and Morton (2011), Glynn and Infanger (2013)). A complete mathematical treatment of these concepts appears under the banner of "Statistical Validation" in Shapiro et al. (2009), and a detailed tutorial for SAA appears in Homem-de Mello and Bayraksan (2014). Since SAA is a batch-oriented method, one requires the choice of a sample size to be used by the SO model. If the solution obtained for a given sample is deemed as statistically unacceptable, then one increases the sample size, and repeats this entire process. The use of a deterministic optimization algorithm within SAA, leads to a method with strict delineation between statistical analysis and optimization. While this separation is convenient from the point of view of implementation, its algorithmic realization can be computationally intensive because batch-oriented methods are not designed to conveniently accommodate increases in batch size. Nevertheless, there have been theoretical investigations on how the computational budget ought to be allocated so that increases in sample size can be determined in an online manner (e.g., Royset and Szechtman (2013)). For problems of practical size, the inability to resume SAA using previously discovered optimization information (e.g. algorithmic quantities such as estimated subgradients) ends up being a serious handicap. In contrast, online methods (or internal sampling schemes of SP) methods are, by their very nature, capable of adapting to increases in sample size. For stochastic programming, online methods such as stochastic approximation (SA) (Nemirovski et al. (2009)) or stochastic decomposition (SD), (Higle and Sen (1991), Higle and Sen (1994)) can easily accommodate online increases in sample size when the "δ-optimality" requirement is not satisfied. In the following we show how one can obtain solutions with statistical guarantees using concepts of statistical optimality. To accomplish this goal, we will combine the algorithmic framework of Sen and Liu (2016) and Corollary 5.19 of Shapiro et al. (2009) to obtain a distribution-free estimate of the probability of optimality of the proposed solution. In the interest of preparing a self-contained presentation, we provide brief summaries of SAA and SD in the Appendix B.

In the following the assumptions of SD (also stated in Appendix B) are expected to hold. In our notation below, $\nu = 1, ..., M$ will denote the index of replications, and for each ν, the SD algorithm is assumed to run for $K_\nu(\varepsilon)$ samples, to produce a terminal solution $\mathbf{x}^\nu(\varepsilon)$, and a terminal value f_ε^ν, where ε is the stopping tolerance

used for each replication. Recall from Appendix B, the grand-mean approximation $\bar{F}_M(x) := \frac{1}{M}\sum_{\nu=1}^{M} f^\nu(x)$, where $\{f^\nu\}_{\nu=1}^{M}$ denotes terminal value function approximations for each replication m. In addition, $\bar{\mathbf{x}} = (1/M)\sum_\nu x^\nu$, and the compromise solution $\mathbf{x}^c$ is defined by $\mathbf{x}^c \in \arg\min\{\bar{F}_M(x) + \frac{\bar{\rho}}{2}||x - \bar{\mathbf{x}}||^2 : x \in \mathbf{X}\}$, where $\bar{\rho}$ is the sample average of $\{\rho^\nu\}$, which denote the terminal proximal parameter for the ν^{th} replication.

THEOREM 1. Assume $\mathbf{X}$ is non-empty, closed and convex, and the approximations f^ν are proper convex functions over $\mathbf{X}$. For $\delta = \bar{\rho}||\mathbf{x}^c - \bar{\mathbf{x}}||^2$, we have,

$$\frac{1}{M}\sum_{\nu=1}^{M} f_\varepsilon^\nu + \delta \geq \bar{F}_M(\mathbf{x}^c). \tag{6}$$

which implies $\mathbf{x}^c$ is δ-argmin to $\frac{1}{M}\sum_{\nu=1}^{M} f^\nu(\cdot)$, and the tolerance level satisfies $\delta = \bar{\rho}||\mathbf{x}^c - \bar{\mathbf{x}}||^2$.

Proof. Since $\mathbf{x}^c \in \arg\min\{\bar{F}_M(x) + \frac{\bar{\rho}}{2}||x - \bar{\mathbf{x}}||^2 : x \in \mathbf{X}\}$, we have,

$$0 \in \partial\bar{F}_M(\mathbf{x}^c) + \mathcal{N}_X(\mathbf{x}^c) + \bar{\rho}(\mathbf{x^c} - \bar{\mathbf{x}}).$$

Hence, $-\bar{\rho}(\mathbf{x^c} - \bar{\mathbf{x}})$ can be used as a subgradient of the function $\bar{F}_M(x) + \mathcal{I}_X(x)$ at $x = \mathbf{x}^c$. Hence, for all $x \in \mathbf{X}$,

$$\bar{F}_M(x) + \mathcal{I}_X(x) \geq \bar{F}_M(\mathbf{x}^c) + \mathcal{I}_X(\mathbf{x}^c) - \bar{\rho}(\mathbf{x}^c - \bar{\mathbf{x}})^\top(x - \mathbf{x}^c)$$

Since $\bar{\mathbf{x}}, \mathbf{x^c} \in \mathbf{X}$, the indicator terms vanish, and therefore,

$$\bar{F}_M(\bar{\mathbf{x}}) + \bar{\rho}(\mathbf{x}^c - \bar{\mathbf{x}})^\top(\bar{\mathbf{x}} - \mathbf{x^c}) \geq \bar{F}_M(\mathbf{x}^c).$$

Since $\bar{\rho}(\mathbf{x^c} - \bar{\mathbf{x}})^\top(\bar{\mathbf{x}} - \mathbf{x^c}) \leq \bar{\rho}||\mathbf{x^c} - \bar{\mathbf{x}}||\ ||\bar{\mathbf{x}} - \mathbf{x^c}||$, we have

$$\bar{F}_M(\bar{\mathbf{x}}) + \bar{\rho}||\mathbf{x^c} - \bar{\mathbf{x}}||^2 \geq \bar{F}_M(\mathbf{x}^c). \tag{7}$$

Recall that $\bar{\mathbf{x}} = \frac{1}{M}\sum_\nu \mathbf{x}^\nu$, and $\bar{F}_M$ is convex, therefore, $F_M(\bar{\mathbf{x}}) \leq \frac{1}{M}\sum_\nu \bar{F}_M(\mathbf{x}^\nu)$. Because $f^j(\mathbf{x}^\nu) \leq f^\nu(\mathbf{x}^\nu)$ for all pairs (j, ν), and $f_\varepsilon^\nu = f^\nu(\mathbf{x}^\nu)$, we have

$$\bar{F}_M(\bar{\mathbf{x}}) \leq \frac{1}{M}\sum_\nu \bar{F}_M(\mathbf{x}^\nu) \leq \frac{1}{M}\sum_\nu f_\varepsilon^\nu. \tag{8}$$

Combining (7) and (8), we get

$$\frac{1}{M}\sum_\nu f_\varepsilon^\nu + \delta = \frac{1}{M}\sum_\nu f_\varepsilon^\nu + \bar{\rho}||\mathbf{x^c} - \bar{\mathbf{x}}||^2 \geq \bar{F}_M(\mathbf{x}^c). \quad \blacksquare$$

If we define $\hat{S}_M(\delta) = \{x \in \mathbf{X} \mid \bar{F}_M(x) \leq \frac{1}{M}\sum_\nu f_\nu(\mathbf{x}^\nu) + \delta\}$, f^* the optimal value, and $S(\delta_u) = \{x \in \mathbf{X} \mid \bar{F}_M(x) \leq f^* + \delta_u\}$, then Theorem 1 has proved that $\mathbf{x}^c \in \hat{S}_M(\delta)$. Note that $S(\delta_u)$ defines the solution set which is δ_u-optimal to the true optimal solution, we should also analyze the relationship between $\mathbf{x}^c$ and $S(\delta_u)$.

Unless one restricts the model to using only an Empirical Distribution (ED), it is difficult for a user to prescribe a sample size for a stochastic optimization model. Hence we do not recommend this for cases where the distribution used is continuous or discrete with countably infinite number of outcomes. Instead, we use SD to suggest sample sizes, and discover the probability that a recommendation $\mathbf{x}^c \in S(\delta_u)$. The following theorem gives the probability bound of $\mathbf{x}^c \in S(\delta_u)$.

THEOREM 2. Let $F(x,\tilde{\xi}) := c(x) + H(x,\tilde{\xi})$ denote the objective rv in (2) and (4). Suppose for each outcome ξ, $\kappa(\xi)$ satisfies $|F(x',\xi) - F(x,\xi)| \leq \kappa(\xi)||x' - x||$. We define the Lipschitz constant of $\mathbb{E}_\xi[F(x,\tilde{\xi})]$ as $L = \mathbb{E}_\xi[\kappa(\tilde{\xi})]$. Suppose $\mathbf{X} \subseteq \mathbb{R}^n$ has a finite diameter D, and let the tolerance level $\delta_u > \delta$, with δ defined in Theorem 1. Then we have the following inequality:

$$Prob(\hat{S}_M(\delta) \subset S(\delta_u)) \geq 1 - \exp\left(-\frac{NM(\delta_u - \delta)^2}{32L^2D^2} + n\ln\left(\frac{8LD}{\delta_u - \delta}\right)\right). \tag{9}$$

Proof. Most SP algorithms which compute subgradients should be able to calculate the Lipschitz constants during algorithm execution, therefore L can be obtained from the solver. If we solve for the probability from (22) in Proposition 1, the following inequality holds:

$$Prob(\hat{S}_M(\delta) \subset S(\delta_u)) \geq 1 - \exp\left(-\frac{K(\delta_u - \delta)^2}{8\lambda^2D^2} + n\ln\left(\frac{8LD}{\delta_u - \delta}\right)\right). \tag{10}$$

From assumption A4-c in Appendix B, $\lambda = 2L$. Also, recall from (23) in Appendix B, each replication uses a sample size of at least N. Therefore, in this case the total sample size K is at least NM. The conclusion holds by replacing λ and K in (10). ■

One of the main strengths of adopting the vision of statistical optimality is the ability to run solution algorithm in an adaptive manner in which iterations with larger sample sizes can be undertaken without having to restart the algorithmic process from scratch. The SD algorithm for SLP problems adopts this strategy.

Remark 3. To the best of our knowledge, the sample size formulas for SAA (Chapter 5, Shapiro et al. (2009)) are not intended for use to set up computational instances for

solution. Instead, their primary role has been in showing that the growth of sample size for SAA depends logarithmically on the size of the feasible set and the reliability level $(1-\alpha)$. Our approach, seeking probabilistic bounds, allows us to compute the reliability of a solution for a sampling-based algorithm (where the probability space corresponds to the product probability measure). The computational results reported in this paper provide such bounds for all included instances. This is made possible by an algorithmic design whereby the sample size is not a pre-requisite for a sequential sampling algorithm. ■

4. Model Validation, Assessment and Selection

The field of statistics, and more recently, Statistical Learning have developed notions of model selection on the basis of estimated errors for models which use empirical distributions. Because of their data driven emphasis, concepts such as model assessment and selection are important for LEO as well. The stochastic optimization (SO) literature has some foundational results for assessing solution quality as proposed in Mak et al. (1999). Shapiro and Homem-de Mello (1998) and Higle and Sen (1996a). However, these tests are not proposed within the larger context of model validation and assessment. Because the LEO setup includes both statistical modeling as well as optimization, we have the potential for both model validation, assessment and selection.

The protocol we adopt is one based on Figure 2b where validation is critical part of the modeling process. These validity tests are embodied in the diamond-shaped blocks of Figure 2b. In section 4.1, we consider the identification of outliers as a data preprocessing step before optimization as in the diamond-shaped block following the statistical model in Figure 2b. Note that this only depends on the dataset and not any decision choice x. In section 4.2 we discuss metrics for any LEO model, and comparisons between alternative LEO models will be presented in section 4.3. These tests correspond to the hypothesis tests used in the lower diamond-shaped block of Figure 2b, and require a decision as an input.

4.1. Data Preprocessing

In stating the LEO model, the class of regressions $\mathcal{M}$ can be quite general. However, a model with Shared Spaces may call for a constrained regression where $\mathcal{M}$ may include bounds on predictions. For instance, in the LEO-Wyndor example, an unconstrained regression may lead to predictions which violate the bounds of the data-set. To identify outliers, we assume that the range of data in learning process should match the range of decision

variables in optimization process for LEO problems. In the LEO-Wyndor example, this assumption indicates that for any decision variable value x in its domain, $\hat{m}(x)$ should be within the observation bound of W from data. Therefore, for models with Shared Spaces, we require some restrictions on the data-set of the learning process.

Let $W_L = \min_i\{W_i : i \in T\}$ and $W_U = \max_i\{W_i : i \in T\}$. Once these bounds W_L, W_U have been computed, we identify those $i \in V$ as outliers by checking whether $m(Z_i, \xi) \in [W_L, W_U]$. Hence, data points with predictors outside the bounds ($[W_L, W_U]$) are considered to be outliers, and should be removed. Figure 3 shows the q-q plots for the error terms of the Training and Validation data sets of the LEO-Wyndor example before and after data preprocessing. We also compared the χ^2 test result of error sets before and after preprocessing. The detailed results of χ^2 test are included in section 5, where all computational results are presented.

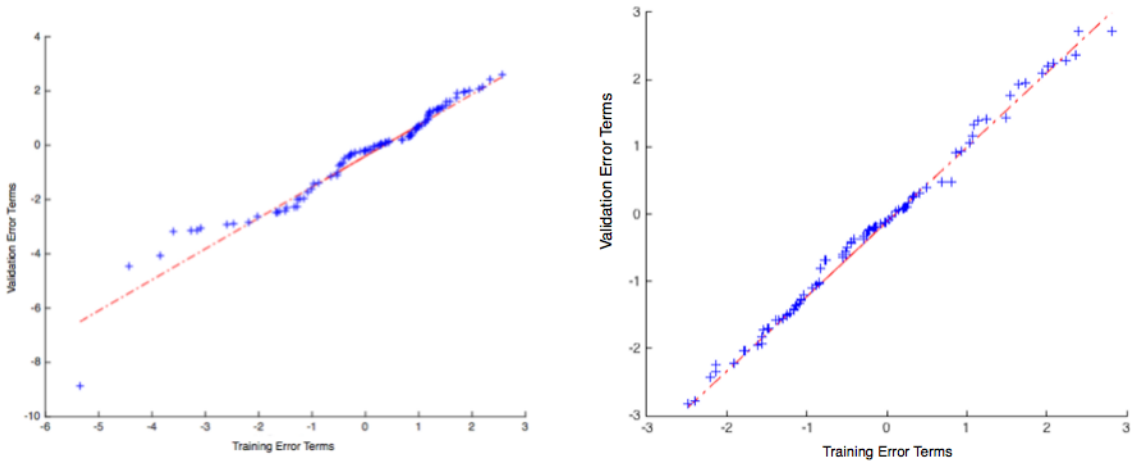

Figure 3 **q-q plot before and after data preprocessing**

4.2. Metrics and Model Validation

The following tests will be included for each alternative LEO model (indicated by an index q are mentioned in section 2).

- χ^2 test for error terms and cost-to-go objectives
- T-test for the mean of cost-to-go function
- F-test for the variance of cost-to-go function

The last two tests are based on asymptotic normality of the optimal value and the optimal solutions of the stochastic optimization problem. The property of asymptotic normality

is obtained via uniform convergence of SAA (Homem-de Mello and Bayraksan (2014)) whereas the asymptotic normality of solutions follows from the uniqueness of limits as shown in Sen and Liu (2016). The latter property does not necessarily hold in the general SAA setting which does not impose any algorithmic conditions. If the null hypothesis is rejected in either case, then the corresponding LEO model is rejected. It is worth noting that these tests can be applied to other classes of optimization models as well. For instance, in section 5 we will study how a standard linear program might perform under these validation tests.

4.2.1. χ^2 Test for Error Terms and Cost-to-go Objectives. We perform χ^2 tests for error terms and the cost-to-go objective functions. To start with, we describe a general χ^2 test. The data-set is expected to have two parts, and we test the null hypothesis that both parts of the data-set share a common distribution.

Given a data-set we allocate the data into B bins, for the i^{th} bin, denote E_{1i} as the observed frequency for bin i from one sample, and E_{2i} as the observed frequency for bin i from validation the other sample. Then the χ^2 statistic for this data is estimated as:

$$\hat{\chi}^2 = \sum_{i=1}^{B} \frac{(E_{1i} - E_{2i})^2}{E_{1i} + E_{2i}} \tag{11}$$

We check the χ^2 distribution with B degrees of freedom which provides the standard value $\chi^2(B)$, and the probability $p = Prob(\chi^2(B) > \hat{\chi}^2)$. Given a significance level α, we reject the null hypothesis if $p \leq \alpha$; otherwise, we do not reject the null hypothesis.

Error Terms. One of our assumptions (Assumption A1-homoscedasticity) is that the error terms are independent of the decisions x. We test the following null hypothesis.

$$H_0: \quad \text{the two data sets } (\{\xi_i\}_{i\in T}, \{\xi_i\}_{i\in V}) \text{ share a common distribution.} \tag{12}$$

If this null hypothesis is rejected, it indicates that the error terms do not satisfy our assumptions and hence the statistical model would be inappropriate.

Cost-to-go Objective. Let $\hat{x}$ denote the solution being evaluated, and $\{h(\hat{x}, \xi_i)\}_{i\in T}, \{h(\hat{x}, \xi_i)\}_{i\in V}$ be defined in equation (3) or (5), depending on whether we have Disjoint or Shared Spaces. Using (11), we test the following null hypothesis.

$$H_0: \quad \text{the two data sets } (\{h(\hat{x}, \xi_i)\}_{i\in T}, \{h(\hat{x}, \xi_i)\}_{i\in V}) \text{ share a common distribution.} \tag{13}$$

4.2.2. Cost-to-go Hypothesis test for the Mean value using T-statistic. For cost-to-go objectives, we introduce the null hypothesis test for the mean value. We will determine the difference of mean values depending on whether the confidence intervals of two samples overlap. Suppose we have two independent samples, the mean values of them are h_1, h_2, and the standard deviations are s_1, s_2. To determine whether two sample means are significantly different with $\alpha = 0.05$, then t-statistic of two group means is

$$t = \frac{|h_1 - h_2|}{\sqrt{s_1^2 + s_2^2}}.$$

Compare the calculated t-value with $t_0 = 1.96$, if $t > t_0$, we will reject the null hypothesis, which indicates that the mean values of two samples are significantly different. If this hypothesis is rejected, the objectives of training and validation set are considered to be different on the basis of the first moment.

4.2.3. Cost-to-go Hypothesis test for the Variance value using F-statistic. Besides the hypothesis test on the first moment level, we also perform a test for the variance value based on the F distribution. The F-test is often used to test if the variances of two samples are consistent. The null hypothesis H_0 is defined as: the variances of two samples are equal. Suppose we denote the observed variances of two samples as s_1^2 and s_2^2, then the F statistic is the following:

$$F = \frac{s_1^2}{s_2^2}$$

Suppose we choose the significance level α, sample 1 has sample size N_1, and sample 2 has sample size N_2, then the critical region is decided by two values from F-distribution: $F_{\alpha/2,N_1-1,N_2-1}, F_{1-\alpha/2,N_1-1,N_2-1}$. If $F_{\alpha/2,N_1-1,N_2-1} \leq F \leq F_{1-\alpha/2,N_1-1,N_2-1}$, we do not reject the null hypothesis.

4.3. Comparison across LEO Models

In this subsection, we discuss how alternative LEO models are assessed and which of these should be recommended as the most appropriate. In order to do so, we first estimate generalization error and optimization error. Finally, we include the Kruskal-Wallis test for a non-parametric one-way ANOVA, which provides a sense of reliability of the estimates.

Generalization Error. This quantity is a prediction of out-of-sample cost-to-go error which may be observed when the system is implemented in practice. Observe that the

expected "in-sample cost-to-go error" must be explained by the sum of the expected "cost-to-go training error" and the expected "out-of-sample cost-to-go error". Accordingly, the expected out-of-sample cost-to-go error can be estimated by the difference between two quantities: one is the unobservable future cost-to-go and the training cost-to-go data, and the other is the validation cost-to-go data and the training cost-to-go data. Let the in-sample cost-to-go error be approximated as

$$\mathrm{Err}_{in} \approx \frac{1}{|T|}\sum_{i=1}^{|T|}\mathbb{E}_{H^+}(H_i^+ - \hat{h}_i)^2, \tag{14}$$

where H_i^+ represents a new observation of the cost-to-go function, and $\hat{h}_i$ denotes a cost-to-go function value in the training dataset of sample size $|T|$. Thus, the in-sample cost-to-go error estimates an average error between a new cost-to-go response and the training set cost-to-go.

Let h_i represent the validation cost-to-go objective, and the cost-to-go training error (err) is defined as

$$\mathrm{err} = \frac{1}{|T|}\sum_{i=1}^{|T|}(h_i - \hat{h}_i)^2. \tag{15}$$

Given (14) and (15), the generalization error is estimated by $\mathbb{E}_h(\mathrm{Err}_{in} - \mathrm{err})$. The following theorem suggests a mechanism to estimate generalization error.

THEOREM 3. Assume that the expected value of new observations of the cost-to-go function ($\mathbb{E}_{H^+}H_i^+$) is equal to the expectation of the validated cost-to-go function ($\mathbb{E}_h h_i$), and suppose that assumptions A1 and A2 hold. Then the generalization error is

$$\mathbb{E}_h(\mathrm{Err}_{in}) - \mathbb{E}_h(\mathrm{err}) \approx \frac{2}{|T|}\sum_{i=1}^{|T|}\mathrm{Cov}(h_i, \hat{h}_i) \tag{16}$$

Proof. The following equations hold:

$$\begin{aligned}
\mathbb{E}_h(\mathrm{Err}_{in}) - \mathbb{E}_h(\mathrm{err}) &\approx \frac{1}{|T|}\sum_{i=1}^{|T|}\mathbb{E}_h\mathbb{E}_{H^+}(H_i^+ - \hat{h}_i)^2 - \frac{1}{|T|}\sum_{i=1}^{|T|}\mathbb{E}_h(h_i - \hat{h}_i)^2 \\
&= \frac{1}{|T|}\sum_{i=1}^{|T|}\left[\mathbb{E}_h\mathbb{E}_{H^+}(H_i^{+2} + \hat{h}_i^2 - 2H_i^+\hat{h}_i) - \mathbb{E}_h(h_i^2 + \hat{h}_i^2 - 2h_i\hat{h}_i)\right] \\
&= \frac{1}{|T|}\sum_{i=1}^{|T|}\left[\mathbb{E}_{H^+}H_i^{+2} + \mathbb{E}_h\hat{h}_i^2 - 2\mathbb{E}_h\mathbb{E}_{H^+}(H_i^+\hat{h}_i) - \mathbb{E}_h h_i^2 - \mathbb{E}_h\hat{h}_i^2 + 2\mathbb{E}_h(h_i\hat{h}_i)\right]
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{|T|}\sum_{i=1}^{|T|}\left[\mathbb{E}_{H^+}H_i^{+2} - 2\mathbb{E}_{H^+}\mathbb{E}_h(H_i^+\hat{h}_i) - \mathbb{E}_h h_i^2 + 2\mathbb{E}_h(h_i\hat{h}_i)\right] \\
&= \frac{1}{|T|}\sum_{i=1}^{|T|}\left[2\mathbb{E}_h(h_i\hat{h}_i) - 2\mathbb{E}_h(h_i)\mathbb{E}_h(\hat{h}_i)\right] \\
&= \frac{2}{|T|}\sum_{i=1}^{|T|}\mathrm{Cov}(h_i,\hat{h}_i),
\end{aligned}
$$

where the first equation is a result of (14) and (15), the second and third are due to algebraic manipulcations, the fourth follows from assumption A2 that $\mathbb{E}_{H^+}H_i^{+2} = \mathbb{E}_h h_i^2$, and the fifth by definition. ■

Therefore, the covariance above is an estimate of the generalization error. Among alternative models, if we observe one with a larger covariance than another, then we may conclude that the one with a lower covariance has a lower generalization error.

As is common in statistical learning, one obtains better estimation of generalization error by using cross-validation (Hastie et al. (2011)). In one run of cross-validation, the data is partitioned randomly into two complementary subsets. To analyze the generalization error for a given decision, we calculate the covariance of cost-to-go objectives from these independent subsets. Multiple runs of cross-validation will be performed to sample a generalization error, and finally, we report the estimate of the generalization error as the average value over k runs.

Kruskal-Wallis test (Non-parametric One-way ANOVA on ranks). To choose an optimum from all the decisions, we need to find a proper metric to compare the estimated objectives among different models. In this case, we propose to undertake the Kruskal-Wallis test, which does not assume normality as a condition for the test. The null hypothesis of Kruskal-Wallis test is that the ranked medians of bins (of samples from two competing models) are the same. When the hypothesis is rejected, the cost-to-go values of one method stochastically dominates the cost-to-go of the other method.

Optimization Error. Suppose that the optimal value of the best model is estimated as $\hat{f}^*$. This value can be obtained by choosing the best model identified via the Kruskal-Wallis test, which will be performed by the pairwise comparisons. Let Q denote the index set of alternative LEO models being compared. Then identifying the best model is accomplished by carrying out $\frac{|Q|(|Q|-1)}{2}$ hypothesis tests. Once $\hat{f}^*$ is identified by these tests, we calculate

the optimization error by the difference $|\hat{f}_q - \hat{f}^*|$, where $\hat{f}_q$ denotes the estimated cost provided by model q.

5. Illustrative Computations

In this section, we describe the work-flow and computational results for two applications, one for models with Disjoint Spaces (LEO-`ELECQUIP`), and another for models with Shared Spaces (LEO-Wyndor).

5.1. LEO-`ELECEQUIP`

In this example, we use $c_u = 1, c_v = 3$ and $U_t = R_t = \infty$ (see Appendix A for the notation).

(a) Deterministic ARIMA Forcasting (DAF). Since U_t and R_t are infinity, we can use the predicted demand to define the order quantity as: $\Delta_t = \text{Max}\{0, \hat{D}_t - u_t\}$, where $\hat{D}_t$ is the expected value of the ARIMA model.

(b)Stochastic Linear Programming (SLP), which gives the decision by solving the problem in equation (18) in Appendix A. Note that our rolling horizon approach solves three period problems (0,1,2), and we use the solution of period 0 as our current decision, and drop the other decisions. We then use the demand of the following period, update the inventory status, and move the clock forward to the next period.

5.1.1. Month by Month Validation Results for 2001-2002. The ARIMA model was trained on data from 1996-2000, and the performance of the models were validated during the two year period 2001-2002. Table 1 presents the costs for the year 2001 and 2002 (24 months) for each of the two inventory policies in a dynamic view. Note that of the 24 runs (simulating two years of inventory), the LEO approach only lost once, for month 1. Thereafter, it cost less in each subsequent month, with some (months) reducing costs by over 66%. The average inventory cost reduction over the deterministic ARIMA forecast is approximately 34% over the 2 year run.

5.1.2. Snapshot Statistical Comparisons. To illustrate the application of our model validation and assessment statistics (section 4), we select the end of the first year as the point in time when statistical comparisons are made. For this snapshot study, such a choice, allowing the model to run for a year, helps to avoid initialization bias. Table 2 provides the estimated objective, validated objective, and standard deviation of validated cost-to-go objectives. The probability of optimality reported in Table 2 is a result of the computations

Month	1	2	3	4	5	6
DAF Cost	12.33	14.41	39.02	14.54	26.44	28.86
SLP Cost	16.53	3.06	12.28	9.49	20.63	17.77
Month	7	8	9	10	11	12
DAF Cost	7.65	37.99	25.26	38.46	16.92	30.34
SLP Cost	7.38	31.27	14.82	28.66	13.23	21.92
Month	13	14	15	16	17	18
DAF Cost	11.35	3.05	15.11	26.74	15.67	38.98
SLP Cost	6.04	1.11	11.06	15.78	14.22	24.56
Month	19	20	21	22	23	24
DAF Cost	33.23	23.81	17.90	16.62	15.31	29.72
SLP Cost	11.90	19.88	5.13	8.66	9.05	23.20

Table 1 LEO-ELECQUIP: Monthly Back-Testing Costs

suggested in Theorem 2, where δ_u is chosen to be 1% of the total cost. Notice that for the DAF model, we do not report a probability because it is simply a result of the ARIMA forecast. On the other hand, we include the probability for the SLP model, and this is consistent with Remark 2 and statistical optimality of section 3.

Table 3 summarizes results for three hypothesis tests for both DAF and SLP cases. A hypothesis test rejects the null hypothesis at the 95% level when the statistic lies outside the range provided in the table. Upon examining the entries for the t-test, the null hypothesizes for both DAF and SLP are not rejected. We also perform the f-test for DAF and SLP, and the hypothesis of DAF is rejected, implying that the variances of training and validation objective sets are considered to be significantly different at 95%. The results of the χ^2 test are presented in the last two rows, which analyzes the consistency of two data sets. Note that both DAF and SLP are not rejected at level $\alpha = 0.05$, but SLP shows a higher p-value. From these test results, we conclude that the SLP approach performs better in so far as consistency between training and validation sets.

The comparison across models is provided in Table 4. The cost of SLP in validation is smaller than DAF by 9.42, and it also shows a smaller generalization error as well. We include the $p-$value of Kruskal-Wallis test between DAF and SLP approaches, and the

Models	DAF	SLP
Estimated Obj.	25.52	22.75
Validated Obj.	30.34	21.92
Std. Dev. of Validated Obj.	3.36	3.14
Probability (γ)		0.9934
Tolerance (δ)		0.092

Table 2 LEO-ELECQUIP: Comparison of Solutions under Alternative Models

result shows that objectives of DAF and SLP methodologies have significantly different ranked medians. Since LEO-ELECEQUIP is a minimization problem, better solutions result in costs that are at the lower end of the horizontal (cost) axis. In this case, the better decision results from SLP, and Figure 4 gives evidence of this conclusion because for all cost levels C, the $Prob\ (Cost \leq C)$ is higher for SLP than it is for DAF.

Models	DAF	SLP
t-statistic ($t < 1.96$)	$t = 1.21$	$t = 0.20$
Cost-to-go Test (Mean)	not rejected	not rejected
f-statistic ($0.62 < f < 1.62$)	$f = 2.53$	$f = 1.23$
Cost-to-go Test (Variance)	rejected	not rejected
χ^2 Test p-value ($p > 0.05$)	$p = 0.13$	$p = 0.37$
Cost-to-go Test (Distribution)	not rejected	not rejected

Table 3 LEO-ELECQUIP: Hypothesis Test Results under Alternative Models

Models	DAF	SLP
Generalization Error	1.45	0.96
Kruskal-Wallis Test ($p-$value)	1.24×10^{-6}	
Optimization Error	9.42	

Table 4 LEO-ELECQUIP: Errors under Alternative Models

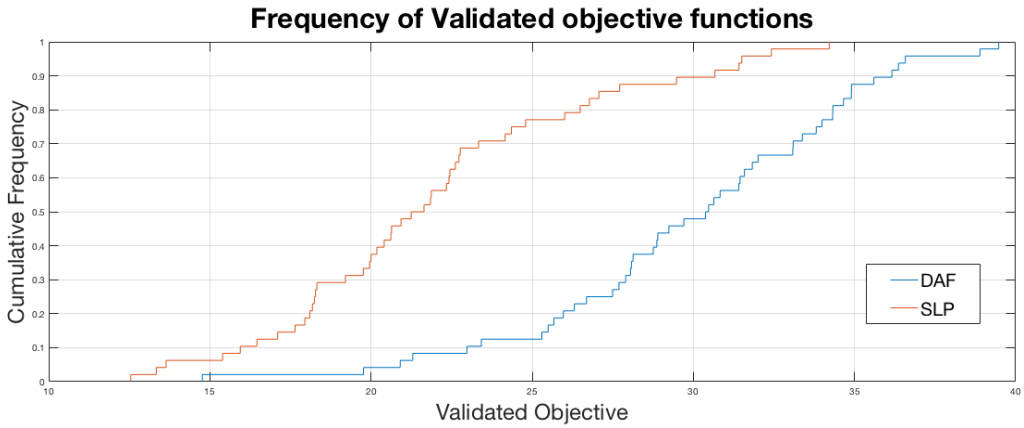

Figure 4 **LEO-ELECQUIP: Stochastic Dominance of SLP Validated objectives over DAF**

5.2. LEO-Wyndor

We also studied the LEO-Wyndor problem (Appendix A) under alternative models. DF/LP represents learning enabled optimization using deterministic forecasts, in which we use the expected value of the linear regression as the demand model. This results in a deterministic LP. In addition, we also study other models where linear regression suggests alternative parameters: a) the additive error model, using the empirical distribution (ED) uses scalar errors and deterministic model coefficients $\beta_0, \beta_1, \beta_2$ where the first is the constant term, the second is the coefficient for TV expenditures, and the third is the coefficient for radio expenditures; b) a linear regression whose coefficients are random variables $\tilde{\beta}_j$, which are normally distributed and uncorrelated (NDU); c) a linear regression whose coefficients are random variables $\tilde{\beta}_j$ which are normally distributed and correlated (NDC). We reiterate that all three models ED, NDU, NDC correspond to specific types of errors (as discussed in section 2). Note that for models NDU and NDC, we have continuous rvs, and as a result we adopted SD as the solution methodology and refer to the results by NDU/SD and NDC/SD. Also note that for the case of ED, the dataset is finite and reasonably manageable. Hence we will use both SAA and SD for this model, and refer to them by ED/SAA and ED/SD.

5.2.1. Results for Error terms. The calculations begin with the first test as the top diamond block in Figure 2(b). Table 5 shows p-values and test results of χ^2 test for NDU/SD, NDC/SD and ED. From values reported in Table 5, the fit appears to improve when a few of the data points near the boundary are eliminated as suggested in section 4.1 (see Figure 3).

	NDU/SD	NDC/SD	ED
Before Data Preprocessing	0.44, not rejected	0.42, not rejected	0.45, not rejected
After Data Preprocessing	0.59, not rejected	0.57, not rejected	0.78, not rejected

Table 5 LEO-Wyndor: Comparison of Chi-square test

5.2.2. Results for Decisions and Optimal Value Estimates. The decisions, predicted profit, validated profit and probability of optimality are shown in Table 6. The plans produced by each model are given in the first two rows, and the predicted profit appears in the third row. The fourth row reports the 95% confidence interval (CI) of expected profits observed using the validation data set. The last two rows report the probability γ and the corresponding tolerance level δ, which are provided by SD algorithm based on the theorems in section 3. We choose 1% of the mean value of validated objective to be δ_u in Theorem 2. Once again, notice that for both DF/LP and ED/SAA, we do not report any probability because we use a deterministic solver as in (4).

The hypothesis test results for the cost-to-go objectives (the lowest diamond in Figure 2(b)) for each model are reported in Table 7. As described in the previous section, the cost-to-go test for the mean value is considered to be more critical since the objective is intended to minimize the expectation. The hypothesis test checks whether the training and validation datasets for the cost-to-go function have the same mean value. The t-test rejects the DF/LP model. The next two rows give the test results of variance based on f-statistic, and we conclude that none of the models can be rejected. We also performed a χ^2 test for the cost-to-go objectives using the training and validation sets. Again, the DF/LP model was rejected where as the others were not.

Remark 4. The concept of cross-validation (k-fold) is uncommon in the stochastic optimization literature. With $k > 1$, this tool is a computational embodiment of (14), and provides a prediction of the error. Without such cross-validation, it is often likely that model assessment can go awry. For instance, in this example we have observed that if we use $k = 1$, then the ED/SAA model can get rejected although using $k = 5$, the ED/SAA is no longer rejected. This can be attributed to the fact that variance reduction due to $k = 5$-fold cross-validation reduces Type I error (when compared with $k = 1$). ■

Table 8 reports the optimization error, as well as the generalization error for all models. DF/LP shows the largest optimization error, which indicates that it is not an appropriate

model to recommend for this application. On the other hand, NDU/SD and NDC/SD have comparable and relatively small generalization errors. However the optimization errors appear to be significant.

In Table 9 we present the pairwise comparison of Kruskal-Wallis non-parametric ANOVA test. For the tests of DF/LP with other methodologies, the p-values are all smaller than 0.01, which implies that there are significant differences between the median ranks of DF/LP and each of the other four approaches. The p-value comparing NDU/SD and NDC/SD is 0.37, and hence the Kruskal-Wallis test does not reject the null hypothesis. Note that the p-value of ED/SAA & ED/SD is also greater than 0.01, and the null hypothesis is not rejected as well. Thus to summarize, the ANOVA test suggests that one ED/SD and ED/SAA are just as good, and both are better than NDU/SD and NDC/SD, and these are all superior to DF/LP.

The stepped curve in Figure 5 illustrates the ordering discovered by the non-parametric ANOVA test. Note that DF/LP shows significant difference from the other approaches. Moreover, the curves for NDU/SD and NDC/SD are relatively close, whereas ED/SAA and ED/SD are indistinguishable. These similarities were quantified in Table 9 by the fact that the p-values for these comparisons are greater than 0.01. Finally, ED/SAA and ED/SD give the largest objective value, which is also reported in Table 6. LEO-Wyndor example is a profit maximization problem, therefore ED/SAA and ED/SD lead to more profitable decisions since they stochastically dominate the others. The ANOVA test suggests that the difference of ED/SAA and ED/SD is not significant, therefore both ED/SAA and ED/SD provide the most profitable decision (see Table 6 for the validated objective value estimated for ED/SAA and ED/SD).

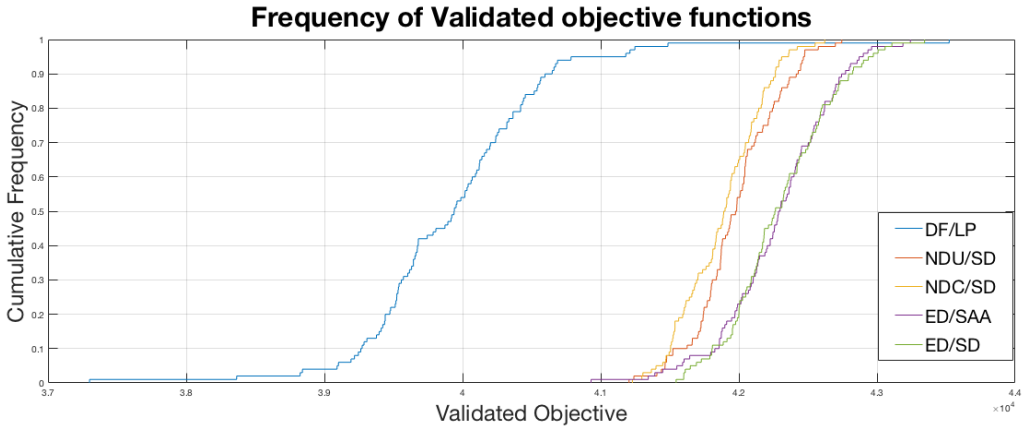

Figure 5 LEO-Wyndor: Stochastic Dominance of Validated Objectives under Alternative Models

Models	DF/LP	NDU/SD	NDC/SD	ED/SAA	ED/SD
x_1	173.48	181.70	181.40	191.27	191.40
x_2	26.52	18.30	18.60	8.73	8.60
Estimated Obj.	\$41,391	\$41,580	\$41,492	\$42,009	\$42,045
Validated Obj. 95% C.I.	\$39,869(±969)	\$41,903 (±468)	\$41,865 (±396)	\$42269 (±718)	\$42,274 (±686)
Probability (γ)		0.9633	0.9698		0.9872
Tolerance (δ)		0.760	0.694		0.842

Table 6 LEO-Wyndor: Comparison of Solutions for Alternative Models

Models	DF/LP	NDU/SD	NDC/SD	ED/SAA	ED/SD
t-statistics($t < 1.96$)	$t = 2.18$	$t = 0.72$	$t = 0.84$	$t = 0.62$	$t = 0.49$
Cost-to-go Test(Mean)	rejected	not rejected	not rejected	not rejected	not rejected
f-statistics($0.67 < f < 1.49$)	$f = 1.23$	$f = 1.43$	$f = 1.29$	$f = 0.79$	$f = 1.16$
Cost-to-go Test(Variance)	not rejected	not rejected	not rejected	not rejected	not rejected
χ^2 Test p-value ($p > 0.05$)	$p = 0.038$	$p = 0.34$	$p = 0.32$	$p = 0.42$	$p = 0.42$
Cost-to-go Test(Distribution)	rejected	not rejected	not rejected	not rejected	not rejected

Table 7 LEO-Wyndor: Hypothesis Test Results for Alternative Models

Models	DF/LP	NDU/SD	NDC/SD	ED/SAA	ED/SD
Optimization Error	2405	371	409	5	
Generalization Error	29.751	19.406	19.554	21.889	21.326

Table 8 LEO-Wyndor: Errors for Alternative Models

Models	ED/SD	ED/SAA	NDC/SD	NDU/SD
DF/LP	2.76×10^{-8}	1.34×10^{-7}	1.12×10^{-7}	5.60×10^{-7}
NDU/SD	8.46×10^{-7}	6.2×10^{-3}	0.37	
NDC/SD	2.05×10^{-7}	1.72×10^{-3}		
ED/SAA	5.87×10^{-2}			

Table 9 LEO-Wyndor: Kruskal-Wallis Test Results ($p > 0.01$)

6. Extensions and Conclusions

The presentation in this paper has focused only on a "batch" (or sequential) version of LEO models, and our illustrations were restricted to models involving linear regression and stochastic linear programming. In this sense, we relegated the vast modern technology of SL and SO into the background. Instead, we focused on presenting a new paradigm for rapid modeling, decision-making based on statistical optimality, and a comprehensive collection of methods for model validation, assessment, and selection. In the interest of broader exposition, we illustrated these concepts in the context of control (inventory) and coordination (production and marketing). These applications are not only widespread in the OR/MS literature, but they also capture two important classes of data types for decision models: namely, cross-sectional data (as in LEO-Wyndor) and time-series data (as in LEO-`ELECEQUIP`). Clearly, there is much to do with a variety of other types of data sets (e.g. spatial, spatio-temporal) and other classes of decision models (e.g. nonlinear, mixed-integer, dynamic and others). We discuss several such extensions below.

(a) More General SL Models. One of the more interesting approaches to regression arises under the banner of projection pursuit regression (PPR). This approach allows regression models to include generalized directions, rather than simply using coordinate directions as in ordinary regression. Because the set of such directions are infinite, it would be best

to interleave regression iterations with optimization iterations. This leads us to the next possible extension: online methods.

(b) Online LEO. The notion of successively interleaving learning and optimization iterations is not only interesting for PPR, but it is also important for instances in which the data set grows periodically, and previously developed models may be updated as more data becomes available. Such advances are particularly critical when it is important for decisions to continue to track a changing environment (as in coordinating renewable energy).

(c) Other Extensions. (i) *Constrained Regression* could be used in cases where points near the boundary of the data set cause inaccuracies. Then the set $\mathcal{M}$ may be modified to include constraints so that the regression coefficients are chosen to satisfy $W_L \leq \hat{m}(Z_i) + \xi_i \leq W_U, i \in T$ in $\mathcal{M}$, where W_L, W_U denote lower and upper bounds on W in the available data set. (ii) *Deep learning algorithms* are often based on using piecewise linear basis functions. If the LEO model has Disjoint Spaces, then, very general regressions can be used. On the other hand, if the LEO model has Shared Spaces, then the first stage approximations would require nonlinear piecewise linear functions leading to Mixed-Integer Programming (MIP) in the first stage approximation.

(d) Multi-objective LEO. Another alternative setup for a non-sequential version is to treat the sub-models of LEO within a multiple objective formulation. Let $(\mu_1, \mu_2) > 0$, and we solve

$$\delta - \text{argmin} \left[\mu_1 \Big\{ \frac{1}{|T|} \sum_{i \in T} \ell(m) \mid m \in \mathcal{M} \Big\} + \mu_2 \Big\{ c(x) + \frac{1}{|T|} \sum_{i \in T} h(x, \xi_{i,q} | Z = z_i), \text{s.t: } x \in \mathbf{X} \Big\} \right]. \tag{17}$$

This formulation is particularly relevant for cases in which μ_2 is the dominant parameter, and the optimization model determines how many iterations are necessary for the statistical model to provide a reasonable approximation for optimum decisions. This form of a LEO model may also be relevant in the context of PPR mentioned above.

(e) LEO Games. A final avenue worth exploring is a multi-agent game in which each agent makes choices based on a multi-objective LEO model where the statistical element reflects previously encountered market observations (supply and demands revealed via a market). In the interest of tractability of such a setup, one might hypothesize that there is a market represented by an econometric model. Such market models may either be commonly shared by all players, or each player may perceive different market models. This approach is likely

to be much more amenable to modeling large scale (and real-time) markets which are currently addressed by Nash-equilibrium models. In the LEO setting, market statistics will become the source of common information, making decisions much more data driven and computationally decentralized.

(f) Software Architecture. The entire workflow (i.e. protocol) of the LEO setup is software intensive because the setting is intended to investigate several plausible statistical models, and their corresponding optimization models. Ultimately, the decisions from these models will be pitted against each other so that the model validation, assessment, and selection procedures will either guide a user or a mechanism to make appropriate choices. Because SO models are usually communicated using algebraic formulations, while SL methods are available through open source software such as "R", we plan to integrate these alternative architectures through a novel open-source platform freely available for academic use. A vision of such software appears in Figure 6. One important observation which we should take away from this figure is that there has been very little progress in visualization for optimization in general, and certainly for SO in particular. The introduction of modern visualization has the potential to help users discover insights through visualization of both SL and SO parts of the LEO.

(g) Parallelization. Since the framework will support using multiple statistical models, it is clear that one should consider using parallel processors, especially in those cases where alternative statistical models are chosen once at the start. This aspect of LEO is clearly low-hanging fruit, and would be extremely important for practitioners.

We close with some key words which summarize this paper: statistical optimality, model fidelity, optimization/generalization error, model selection (and model rejection). Given the numerous directions in which the future can unfold, we invite the optimization community to join in this learning enabled optimization adventure.

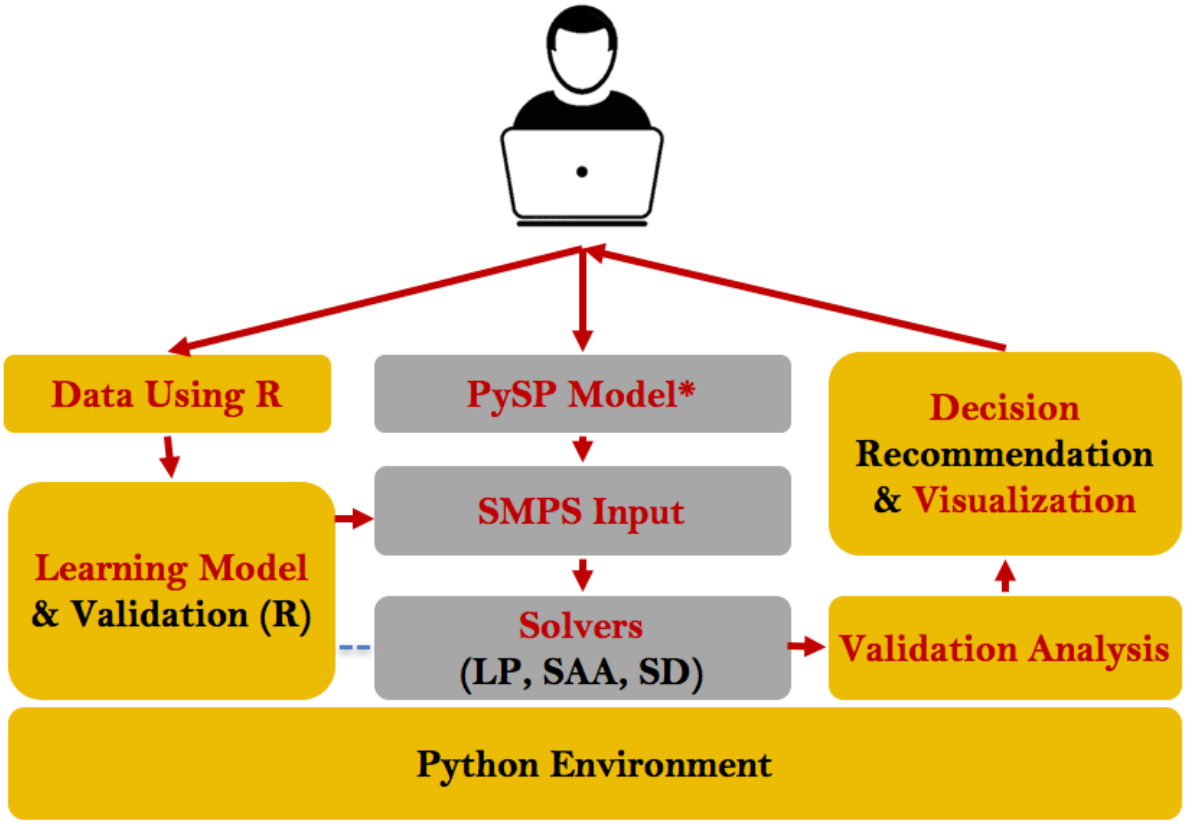

Figure 6 Software Framework (PySP refers to the software described in Watson et al. (2012))

Appendix A: Example LEO Models

The instances discussed below are developed using existing data-sets and existing optimization models. As with the rest of the paper, the novelty here is in the fusion of learning data-sets and optimization models. We include one example for each type of a LEO structure: Disjoint Spaces and Shared Spaces. Since the data-sets are not new, we append the acronym LEO to the names of the existing data-sets.

A.1. A Model with Disjoint Spaces: LEO-ELECEQUIP (Time-Series Data)

This model is based on a "rolling-horizon" decision model commonly used in inventory control. Before starting a sequence of decisions, one typically analyzes historical demand. In this particular example, we use a commonly available data set referred to as ELECEQUIP which provides demand of electrical equipment over a ten-year period. We will use the first five years to discover the time series pattern of demand, and then, use it within a rolling horizon inventory model. We conduct the validation exercise for two years, 2001-2002.

A.1.1. Statistical Learning

When fitting an ARIMA model to the training data, one identifies (p, d, q) and $(P, D, Q)_\tau$ where τ represents the seasonal backshift, and (p, d, q) specify the AR(p), I(d) and MA(q) components of ARIMA. The following procedure provides a useful general approach.

- Let N_0 denote the number of data points in the training set, and plot the data. Identify outliers (or unusual observations), if any.
- Transform the data if necessary.
- Ensure that data are stationary (using differencing if necessary, and conducting the KPSS test). This determines the ARIMA model.

A.1.2 Stochastic Optimization

Model Details: Without loss of generality, we can view the decision epoch as $j = 0$, and the most recent demand will be denoted $d_0 = D_{j+1}$ (from the validation data set). The beginning inventory y_0 and and ending inventory x_0 are also available. The inventory model will look ahead into periods $t = 0, \ldots, T$, , although as in rolling horizon schemes, only Δ_0 will be implemented. The model will be a two-stage multi-period stochastic program with the first stage making ordering decisions $x = (\Delta_0, \ldots, \Delta_{T-1})$, and the second stage predicting the potential cost of the choice Δ_0. As the decision clock moves forward in time, the total cost of inventory management becomes estimated by this process. The various relationships in the inventory model are summarized below.

- Because of the delivery capacity U_t, we must have $0 \le \Delta_t \le U_t$.
- We will sample demand realizations in period t using a forecast $D_t(\omega)$ (using the time series model created in the training phase). Here the notation ω denotes one sample path of demands over the periods $t = 1, \ldots, T-1$. The notation d_0 (in lower case) denotes a deterministic quantity, whereas, the notation $D_t(\omega)$ denotes the outcome ω of the demand (stochastic process) observed in period t.
- We assume that y_0 and d_0 are given. Let $u_t(\omega)$ denote the ending inventory in period t, and $y_{t+1}(\omega)$ denote the beginning inventory in period $t+1$. We have $y_{t+1}(\omega) = u_t(\omega) + \Delta_t$, and a storage (refrigerator) capacity constraint requires that $y_{t+1}(\omega) \le R_{t+1}$, where the latter quantity is given. Then the ending inventory of period t, denoted $u_t(\omega)$, must obey the relationship $u_t(\omega) = \text{Max}\{0, y_t(\omega) - D_t(\omega)\}$. The unit cost of holding inventory is c_u, where $c_u \ge 0$. The total inventory holding cost for period t is then $c_u u_t(\omega)$.
- Let $v_t(\omega)$ denote the lost sales in period t, so that $v_t(\omega) = \text{Max}\{0, D_t(\omega) - y_t(\omega)\}$. Suppose that the per unit cost of lost sales in period t is c_v, where $c_v \ge 0$. Then the total cost of lost sales for period t is $c_v v_t(\omega)$, and the first stage cost is zero.

$$\text{Min } \mathbb{E}_{\tilde{\omega}}\Big[\sum_{t=0}^{T-1} c_u u_t(\tilde{\omega}) + c_v v_t(\tilde{\omega})\Big] \tag{18a}$$

$$\text{s.t.} \quad y_{t+1}(\omega) - u_t(\omega) - \Delta_t = 0 \quad \text{for almost all } \omega \tag{18b}$$

$$y_{t+1}(\omega) \le R_{t+1} \quad \text{for almost all } \omega \tag{18c}$$

$$\Delta_t \le U_t \tag{18d}$$

$$-y_t(\omega) + u_t(\omega) \ge -D_t(\omega) \quad \text{for almost all } \omega \tag{18e}$$

$$y_t(\omega) + v_t(\omega) \ge D_t(\omega) \quad \text{for almost all } \omega \tag{18f}$$

$$u_t(\omega), v_t(\omega), \Delta_t \ge \mathbf{0} \tag{18g}$$

Note that constraints in (18) should be imposed for all possible errors in the training set. However, not all error combinations are sampled, and as result, we say that the constraints must hold for a large enough sample size (which is what we mean by the phrase "almost all" ω). It suffices to say that the sample size used in optimization is decided during the Stochastic Decomposition (SD) algorithmic process, and the exact procedure is beyond the scope of these notes.

A.2. A Model with Shared Spaces: LEO-Wyndor (Cross-Sectional Data for Production - Marketing Coordination)

We study a "textbook"-ish example which has been amalgamated from two textbooks: one on Operations Research (Hillier and Lieberman (2012)) and another on Statistical Learning (James et al. (2013)). Consider a well known pedagogical product-mix model under the banner of "The Wyndor Glass Co." In this example, Hillier and Lieberman (2012) address resource utilization questions arising in the production of high quality glass doors: some with aluminum frames (A), and others with wood frames (B). These doors are produced by using resources available in three plants, named 1, 2, and 3. The data associated with this problem is shown in Table 10

Plant	Prod. time for A (Hours/Batch)	Prod. time for B (Hours/Batch)	Total Hours Available
1	1	0	4
2	0	2	12
3	3	2	18
Profit per Batch	\$3,000	\$5,000	

Table 10 Data for the Wyndor Glass Problem (Hillier and Lieberman (2012))

The product mix will not only be decided by production capacity, but also the potential of future sales. Sales information, however, is uncertain and depends on the marketing strategy to be adopted. Given a budget of \$200,000, the marketing strategy involves choosing a mix of advertising outlets through which to reach consumers. Exercising some "artistic license" here, we suggest that the advertising data set in James et al. (2013) reflects sales resulting from an advertising campaign undertaken by Wyndor Glass. That is, the company advertises both types of doors through one campaign which uses two different media, namely, TV and radio[4]. In our interpretation, product-sales reflect total number of doors sold ($\{W_i\}$) when advertising expenditure for TV is $Z_{i,1}$ and that for radio is $Z_{i,2}$, in thousands of dollars. (This data set has 200 data points, that is, $i = 1, \ldots, 200$). x_1 will denote TV advertising expenditure, and x_2 will denote radio advertising expenditure.

[4] The actual data set discussed in James et al. (2013) also includes newspapers. However we have dropped it here to keep the example very simple.

	TV	Radio	Sales
1	230.1	37.8	22.1
2	44.5	39.3	10.4
3	17.2	45.9	9.3
4	151.5	41.3	18.5
5	180.8	10.8	12.9
...	...	...	...
199	283.6	42	25.5
200	232.1	8.6	13.4

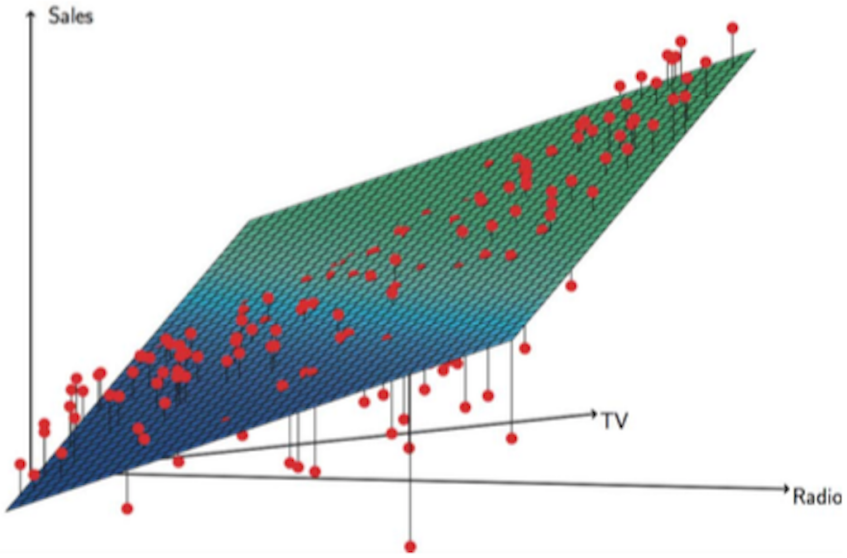

Figure 7 The Advertising Data Set (Source: James et al 2011).

A.2.1. Statistical Learning.

The linear regression model for sales is shown in Figure 7, and will be represented by $\hat{m}(x)$. We consider the following statistical models reported in Section 5.

1. (DF) For deterministic forecasts (DF) we simply use the sales given by $\hat{m}_1(z) = \hat{\beta}_0 + \hat{\beta}_1 z_1 + \hat{\beta}_2 z_2$. Thus, for deterministic predictions, we do not account for errors in forecast, whereas in the case of distributional approximations.

2. (NDU) One approximation of the sales forecast is one where the correlation between the coefficients are ignored, and the resulting model takes the form $m_2(z,\xi) = (\hat{\beta}_0 + \xi_0) + (\hat{\beta}_1 + \xi_1)z_1 + (\hat{\beta}_2 + \xi_2)z_2$, where (ξ_0, ξ_1, ξ_2) are uncorrelated and normally distributed with mean zero, and the standard deviations are computed within MLR.

3. (NDC) Another approximation of the sales forecast is $m_3(z,\xi) = (\hat{\beta}_0 + \xi_0) + (\hat{\beta}_1 + \xi_1)z_1 + (\hat{\beta}_2 + \xi_2)z_2$, where (ξ_0, ξ_1, ξ_2) are correlated and normally distributed according to the variance-covariance matrix reported by MLR.

4. (ED) This is the additive error model, where $m_4(z,\xi) = \hat{\beta}_0 + \hat{\beta}_1 z_1 + \hat{\beta}_2 z_2 + \xi_0$, and ξ_0 denotes a rvs whose outcomes are $W_i - \hat{m}_4(Z_i)$. We refer to this model as empirical distribution.

As in the set up for (4), the index q refers to the alternative error models (DF, NDU, NDC and ED). The corresponding first stage is given as follows:

A.2.2. Stochastic Optimization.

The formulation presented below mimics (4), and since all decisions variables x share the same space as the rv Z, we explicitly remind the reader that $Z = z = x$.

Index Sets and Variables

$i \equiv$ index of product, $i \in \{A, B\}$.

$y_i \equiv$ number of batches of product i produced.

$$x_q \in \delta - \text{argmax } \{-0.1x_1 - 0.5x_2 + \mathbb{E}[\text{Profit}(x, \tilde{\xi}_q \mid Z = z = x)]\} \tag{19a}$$

$$\text{s.t.} \quad x_1 + x_2 \leq 200 \tag{19b}$$

$$x_1 - 0.5x_2 \geq 0 \tag{19c}$$

$$L_1 \leq x_1 \leq U_1, L_2 \leq x_2 \leq U_2 \tag{19d}$$

$$\begin{aligned} \text{Profit}(x, \xi_q \mid Z = z = x) \quad = \quad & \text{Max} \quad 3y_A + 5y_B && \text{(20a)} \\ & \text{s.t.} \quad y_A \leq 4 && \text{(20b)} \\ & \qquad 2y_B \leq 12 && \text{(20c)} \\ & \qquad 3y_A + 2y_B \leq 18 && \text{(20d)} \\ & \qquad y_A + y_B \leq m_q(z, \xi_q) && \text{(20e)} \\ & \qquad y_A, y_B \geq 0 && \text{(20f)} \end{aligned}$$

Note that the choice of ranges $[L_1, U_1]$ and $[L_2, U_2]$ are chosen so that assumption A2 is satisfied. Note that this instance is stated as a "maximization" model, whereas, our previous discussions were set in the context of "minimization". When interpreting the results, it helps to keep this distinction in mind.

Appendix B: Sample Average Approximation (SAA) and Stochastic Decomposition (SD)

Sample Average Approximation(SAA)

Sample Average Approximation is a standard sampling-based SO methodology, which involves replacing the expectation in the objective function by a sample average function of a finite number of data points. Suppose we have sample size of K, an SAA example is as follows:

$$\min_{x\in X} F_K(x) = c^\top x + \frac{1}{K}\sum_{i=1}^{K} h(x,\xi^i). \tag{21}$$

As an overview, the SAA process my be summarized as follows.

1. Choose a sample size K, and collect K outcomes from the training data-set.
2. (Optimization Step). Create the approximation function $F_K(x)$, and solve an SAA instance (21).
3. (Validation Step). Take the decision from $F_K(x)$, follow the steps in section 4, estimate the validated confidence interval, generalization error and optimization error.
4. If the estimated objective does not agree with validated confidence interval, or generalization error and optimization error are not acceptable, increase the sample size K and repeat from step 1.

Assumption 4-a (A4-a). The expectation function $f(x)$ remains finite and well defined for all $x \in \mathbf{X}$. For $\delta > 0$ we denote by

$$S(\delta) := \{x \in \mathbf{X} : f(x) \le f^* + \delta\} \quad \textit{and} \quad \hat{S}_K(\delta) := \{x \in \mathbf{X} : \hat{f}_K(x) \le \hat{f}_K^* + \delta\},$$

where f^* denotes the true optimal objective, and f_K^* denotes the optimal objective to the SAA problem with sample size K.

Assumption 4-b (A4-b). There exists a function $\kappa : \Xi \to \mathbb{R}_+$ such that its moment-generating function $M_\kappa(t)$ is finite valued for all t in a neighborhood of zero and

$$|F(x',\xi) - F(x,\xi)| \le \kappa(\xi)||x' - x||$$

for a.e. $\xi \in \Xi$ and all $x', x \in \mathbf{X}$.

Assumption 4-c (A4-c). There exists constant $\lambda > 0$ such that for any $x', x \in \mathbf{X}$ the moment-generating function $M_{x',x}(t)$ of rv $[F(x',\xi) - f(x')] - [F(x,\xi) - f(x)]$, satisfies

$$M_{x',x}(t) \le exp(\lambda^2||x' - x||^2 t^2/2), \forall t \in \mathbb{R}.$$

From assumption A4-b, $\Big|[F(x',\xi)-f(x')]-[F(x,\xi)-f(x)]\Big| \le 2L||x'-x||$ *w.p. 1, and* $\lambda = 2L$.

PROPOSITION 1. Suppose that assumptions A4(a-c) hold, the feasible set $\mathbf{X}$ has a finite diameter D, and let $\delta_u > 0, \delta \in [0,\delta_u), \alpha \in (0,1)$, and $L = \mathbb{E}[\kappa(\xi)]$. Then for the sample size K satisfying

$$K \ge \frac{8\lambda^2 D^2}{(\delta_u-\delta)^2}\left[n\ln\left(\frac{8LD}{\delta_u-\delta}\right)+\ln\left(\frac{1}{\alpha}\right)\right], \tag{22}$$

we have

$$Prob(\hat{S}_K(\delta) \subset S(\delta_u)) \ge 1-\alpha.$$

Proof: This is Corollary 5.19 of Shapiro et al. (2009) with the assumption that the sample size K is larger than that required by large deviations theory (see 5.122 of Shapiro et al. (2009)). ■

Stochastic Decomposition (SD)

For SLP models, Sen and Liu (2016) have already presented significant computational evidence of the advantage of SD over plain SAA. The reduced computational effort also facilitates replications for variance reduction (VR). VR in SD is achieved by creating the so-called compromise solution which minimizes a grand-mean approximation $\bar{F}_M(x) := \frac{1}{M}\sum_{\nu=1}^{M} f^\nu(x)$, where $\{f^\nu\}_{\nu=1}^M$ denotes a terminal value function approximation for each replication m. Suppose that solutions $x^\nu(\varepsilon) \in (\varepsilon - \arg\min\ \{f^\nu(x) \mid x \in \mathbf{X}\})$ and $\mathbf{x}^c(\delta) \in (\delta - \arg\min\{\bar{F}_M(x) \mid x \in \mathbf{X}\})$. Then, Sen and Liu (2016) have shown consistency in the sense that $\lim_{\delta\to 0}\Pr(\bar{F}_M(\mathbf{x}^c(\delta)) - f^*) \to 0$. Here are some of the critical *assumptions* of SD (Higle and Sen (1996b)).

Assumption 5-a (A5-a). The set of first stage solutions and the set of outcomes are compact.

Assumption 5-b (A5-b).The relatively complete recourse assumption holds.

Assumption 5-c (A5-c).The second stage matrix is deterministic (i.e., fixed recourse).

Assumption 5-d (A5-d).The recourse function h is non-negative. So long as a lower bound on the optimal value is known, we can relax this assumption. (Higle and Sen (1996b))

The value function approximation for replication m is denoted f^ν and the terminal solution for that replication is x^ν. Note that we generate sample average subgradient approximations (SASA) using $K_\nu(\varepsilon)$ observations. Since these observations are i.i.d, the in-sample stopping rule ensures an unbiased estimate of the second stage objective is used for the objective function estimate at x^ν. Hence, the Central Limit Theorem (CLT) implies that $[K_\nu(\varepsilon)]^{\frac{1}{2}}[f(x^\nu) - f^\nu(x^\nu)]$ is asymptotically normal $\mathbf{N}(0, \sigma_\nu^2)$, where $\sigma_\nu^2 < \infty$ denotes the variance of $f^\nu(x^\nu)$. Since

$$N = \min_\nu K_\nu(\varepsilon), \tag{23}$$

it follows that the error $[f(x^\nu) - f^\nu(x^\nu)]$ is no greater than $O_p(N^{-\frac{1}{2}})$. The following result provides the basis for compromise solutions $\mathbf{x}^c$ as proved in Sen and Liu (2016).

PROPOSITION 2. Suppose that assumptions A5(a-d) stated in the Appendix hold. Suppose $\bar{\mathbf{x}}$ is defined as in Theorem 1, and $\mathbf{x}^c = \bar{\mathbf{x}}$. Then,

a) $\mathbf{x}^c$ solves

$$\underset{x \in X}{\text{Min}} \quad \bar{F}_M(x) := \frac{1}{M} \sum_{\nu=1}^{M} f^\nu(x), \tag{24}$$

b)

$$f(\mathbf{x}^c) \leq \bar{F}_M(\mathbf{x}^c) + O_p((NM)^{-\frac{1}{2}}), \tag{25}$$

c) $\mathbf{x}^c(\delta)$ denote an δ-optimal solution to (24). Let f^* denote the optimal value of the problem,

$$\lim_{\delta \to 0} ||\bar{\mathbf{x}}(\delta) - \mathbf{x}^c(\delta)|| \ \to \ 0 \ \text{(wp1)}, \tag{26}$$

d)

$$\lim_{\delta \to 0} P(|\bar{F}_{\delta,N}(\bar{\mathbf{x}}(\delta)) - f^*| \geq t) \ \to \ 0 \ \text{ for all } t \geq 0. \tag{27}$$

Proof: See Sen and Liu (2016).

Acknowledgments

This paper was prepared, in part, for the INFORMS Plenary Lecture given in Nashville, TN, Nov. 15, 2016. This research was supported by AFOSR Grant FA9550-15-1-0267, NSF Grant ECCS 1548847 and NSF Grant CMMI 1538605. We also thank Gabe Hackebeil for extending PySP functionality to allow SMPS output which is necessary for our SD code (available from the authors as well as the Github repository).

References

Bayraksan, Güzin, David P Morton. 2011. A sequential sampling procedure for stochastic programming. *Operations Research* **59**(4) 898–913.

Bertsekas, D.P. 2012. *Dynamic Programming and Optimal Control*. No. v. 2 in Athena Scientific optimization and computation series, Athena Scientific.

Bertsimas, Dimitris, Nathan Kallus. 2014. From predictive to prescriptive analytics. *arXiv preprint arXiv:1402.5481* .

De Farias, Daniela Pucci, Benjamin Van Roy. 2004. On constraint sampling in the linear programming approach to approximate dynamic programming. *Mathematics of operations research* **29**(3) 462–478.

Diaconis, Persi, Mehrdad Shahshahani. 1984. On nonlinear functions of linear combinations. *SIAM Journal on Scientific and Statistical Computing* **5**(1) 175–191.

Frazier, Peter. 2012. Optimization via simulation with bayesian statistics and dynamic programming. *Proceedings of the Winter Simulation Conference*. Winter Simulation Conference, 7.

Gangammanavar, Harsha, Suvrajeet Sen, Victor M Zavala. 2016. Stochastic optimization of sub-hourly economic dispatch with wind energy. *IEEE Transactions on Power Systems* **31**(2) 949–959.

Glynn, Peter W, Gerd Infanger. 2013. Simulation-based confidence bounds for two-stage stochastic programs. *Mathematical Programming* **138**(1-2) 15–42.

Hastie, Trevor J.., Robert John Tibshirani, Jerome H Friedman. 2011. *The elements of statistical learning: data mining, inference, and prediction*. Springer.

Higle, Julia L, Suvrajeet Sen. 1991. Stochastic decomposition: An algorithm for two-stage linear programs with recourse. *Mathematics of Operations Research* **16**(3) 650–669.

Higle, Julia L, Suvrajeet Sen. 1994. Finite master programs in regularized stochastic decomposition. *Mathematical Programming* **67**(1-3) 143–168.

Higle, Julia L, Suvrajeet Sen. 1996a. Duality and statistical tests of optimality for two stage stochastic programs. *Mathematical Programming* **75**(2) 257–275.

Higle, Julia L, Suvrajeet Sen. 1996b. *Stochastic Decomposition*. Springer.

Hillier, Frederick S, G J Lieberman. 2012. *Introduction to operations research*. Tata McGraw-Hill Education.

Homem-de Mello, Tito, Güzin Bayraksan. 2014. Monte carlo sampling-based methods for stochastic optimization. *Surveys in Operations Research and Management Science* **19**(1) 56–85.

James, Gareth, Daniela Witten, Trevor Hastie, Robert Tibshirani. 2013. *An introduction to statistical learning*, vol. 6. Springer.

Kleywegt, Anton J, Alexander Shapiro, Tito Homem-de Mello. 2002. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization* **12**(2) 479–502.

Liyanage, Liwan H, J George Shanthikumar. 2005. A practical inventory control policy using operational statistics. *Operations Research Letters* **33**(4) 341–348.

Mak, Wai-Kei, David P Morton, R Kevin Wood. 1999. Monte carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters* **24**(1) 47–56.

Nemirovski, Arkadi, Anatoli Juditsky, Guanghui Lan, Alexander Shapiro. 2009. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization* **19**(4) 1574–1609.

Olson, Arne, Amber Mahone, Elaine Hart, Jeremy Hargreaves, Ryan Jones, Nicolai Schlag, Gabriel Kwok, Nancy Ryan, Ren Orans, Rod Frowd. 2015. Halfway there: Can california achieve a 50% renewable grid? *IEEE Power and Energy Magazine* **13**(4) 41–52.

Powell, Warren B. 2011. *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, vol. 842. John Wiley & Sons.

Royset, Johannes O, Roberto Szechtman. 2013. Optimal budget allocation for sample average approximation. *Operations Research* **61**(3) 762–776.

Rudin, Cynthia, Gah-Yi Vahn. 2014. The big data newsvendor: Practical insights from machine learning. *DSpace* .

Ryzhov, Ilya O, Warren B Powell, Peter I Frazier. 2012. The knowledge gradient algorithm for a general class of online learning problems. *Operations Research* **60**(1) 180–195.

Sen, Suvrajeet, Yifan Liu. 2016. Mitigating uncertainty via compromise decisions in two-stage stochastic linear programming: Variance reduction. *Operations Research* **64**(6) 1422–1437.

Sen, Suvrajeet, Lihua Yu, Talat Genc. 2006. A stochastic programming approach to power portfolio optimization. *Operations Research* **54**(1) 55–72.

Shapiro, Alexander, Darinka Dentcheva, Andrzej Ruszczynski. 2009. Lectures on stochastic programming. *SIAM, Philadelphia* **10**.

Shapiro, Alexander, Tito Homem-de Mello. 1998. A simulation-based approach to two-stage stochastic programming with recourse. *Mathematical Programming* **81**(3) 301–325.

Watson, Jean-Paul, David L Woodruff, William E Hart. 2012. Pysp: modeling and solving stochastic programs in python. *Mathematical Programming Computation* **4**(2) 109–149.